

BAB IV ANALISA DAN PERANCANGAN

4.1. ANALISA MODEL INTEROPERABILITAS

Pendekatan yang dilakukan berdasarkan literatur contoh Peta Solusi Aplikasi e-Government sebagaimana telah diterangkan pada landasan teori Gambar 2.5., merupakan salah satu alasan pemilihan studi kasus dalam penelitian ini. Sistem informasi kependudukan merupakan dasar dari seluruh pengolahan informasi e-Government, karena memiliki hubungan sektoral yang terkait langsung dengan penduduk atau sumberdaya manusia. Salah satu kasus yang memiliki ketergantungan dengan informasi kependudukan adalah pengolahan data pada sistem informasi perpajakan, khususnya mengenai informasi wajib pajak perorangan.

Berdasarkan skema data wajib pajak yang terdapat dalam Surat Keterangan Terdaftar Wajib Pajak dan skema data penduduk yang terdapat dalam Kartu Tanda Penduduk, sebagaimana terlampir pada tulisan ini, kedua skema data tersebut memiliki irisan skema data yang bermuara pada sebagian struktur yang ada pada Kartu Tanda Penduduk. Hasil irisan skema data tersebut yang dijelaskan dalam Tabel 4.1., menunjukkan adanya keterkaitan antara Sistem Informasi Perpajakan dan Kependudukan. Dalam hal ini proses pertukaran informasi antar kedua Sistem Informasi tersebut dapat dilakukan berdasarkan peta relevansi informasi. Sehingga dibutuhkan sebuah metode pertukaran informasi dalam bentuk Model Interoperabilitas antar Sistem Informasi Perpajakan dan Kependudukan.

Tabel 4.1. Skema Data Penduduk

Field	Type	Comments
nama	text	Nama lengkap
alamat	text	Alamat
kelurahan	text	Kelurahan
kecamatan	text	Kecamatan
kota	text	Kota

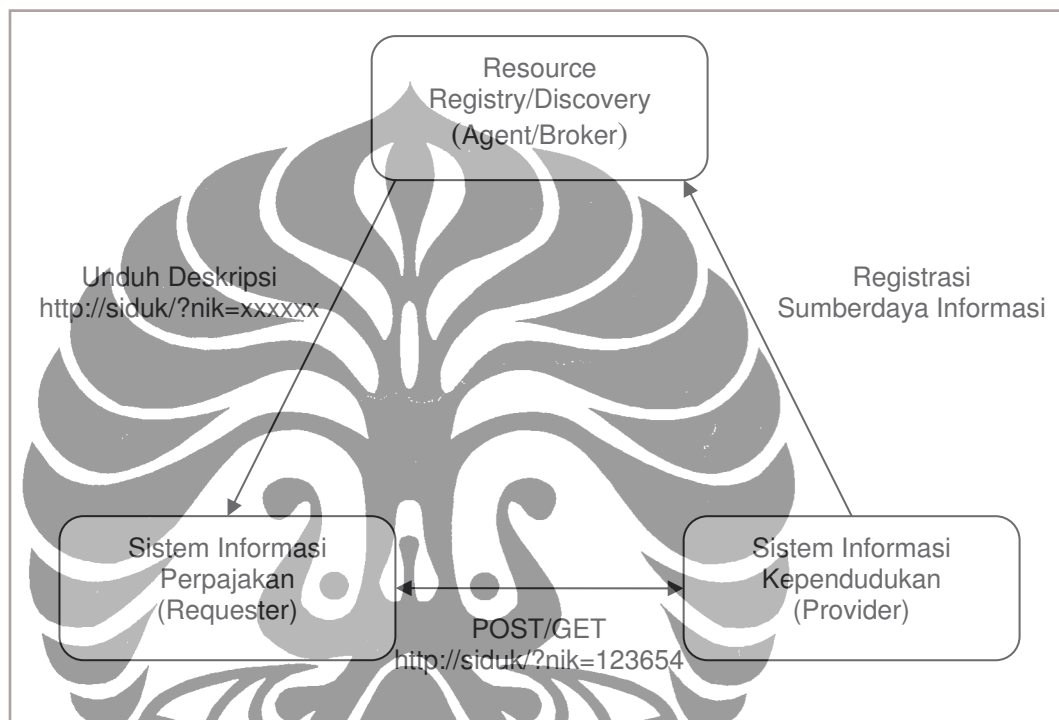
Pada Sistem Informasi Perpajakan terutama mengenai wajib pajak perorangan, proses yang menjadi perhatian adalah validasi dan pengambilan data perorangan untuk wajib pajak baru berdasarkan NIK dapat dilakukan secara otomatis dari data penduduk yang tersedia pada Sistem Informasi Kependudukan. Dengan demikian pada Sistem Informasi Perpajakan tidak diperlukan lagi proses pengisian manual yang terkait dengan data penduduk berupa nama, alamat dan lainnya. Selain itu kedua Sistem Informasi tersebut dapat melakukan proses pertukaran informasi berupa penayangan informasi data penduduk yang sudah dan belum memiliki NPWP.

Hambatan yang dihadapi kedua Sistem Informasi tersebut untuk melakukan proses pertukaran informasi adalah tidak adanya koneksi langsung antar Sistem Informasi karena Sistem Informasi Perpajakan dan Kependudukan berada pada lokasi yang berbeda dengan platform yang dianggap berbeda pula. Maka salah satu solusinya adalah masing-masing Sistem Informasi dikomunikasikan melalui interface pada infrastruktur internet berbasis Web. Interface atau antarmuka yang dijadikan jembatan penghubung antar Sistem Informasi melalui arsitektur Web tersebut adalah Remote Procedure Call (RPC) berbasis Web. Agar proses interkoneksi dan pertukaran data antar Sistem Informasi dapat dilakukan secara sistematis, maka diperlukan Model Interoperabilitas antar Sistem Informasi dengan menggunakan arsitektur Web Services.

Melalui model arsitektur Web Services ini masing-masing Sistem Informasi dipetakan menurut fungsi dan perannya dalam proses pertukaran informasi. Menurut arsitekturnya Web Services memiliki tiga entitas yang berperan dalam proses pertukaran data atau informasi, yaitu Provider sebagai penyedia sumberdaya informasi, kemudian Agent/Broker yang bertugas mengelola dan menyediakan fasilitas untuk registrasi, publikasi, dan penemuan sumberdaya informasi (layanan Provider). Dan terakhir adalah Requester, sebagai pengguna akses sumberdaya informasi yang diberikan oleh Provider.

Untuk kasus validasi dan pengambilan data penduduk oleh Sistem Informasi Perpajakan, maka yang berperan sebagai Provider adalah Sistem Informasi Kependudukan, dan yang berperan sebagai Requester adalah Sistem

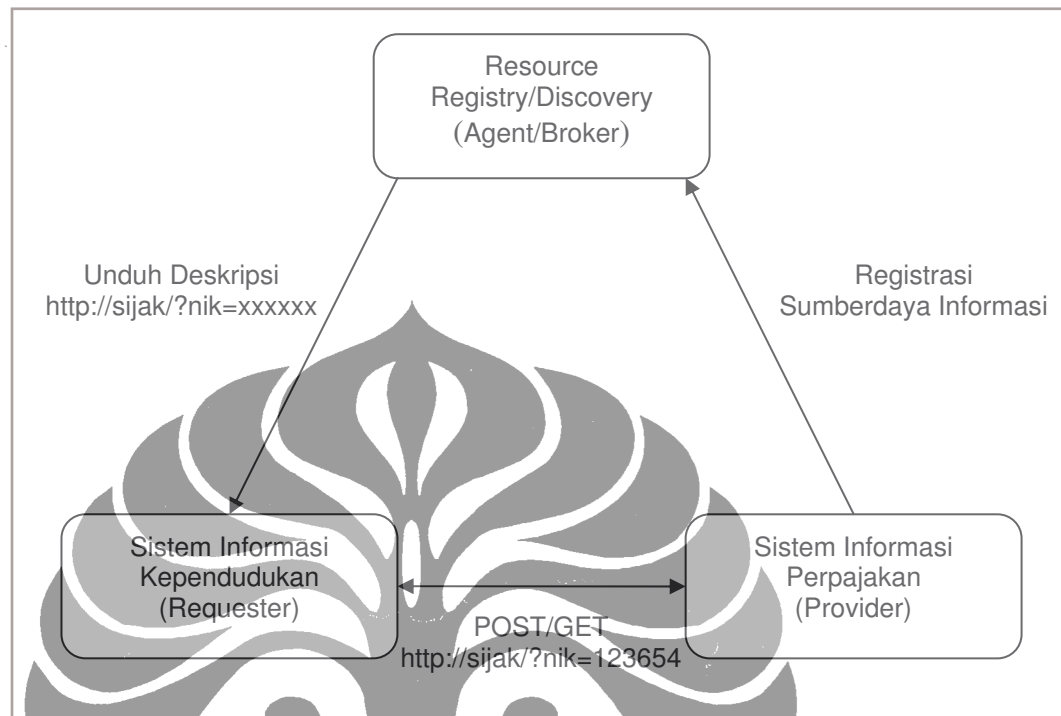
informasi Perpajakan. Sementara itu Agent/Broker dapat diasumsikan sebagai jasa layanan publik pihak ketiga, namun tidak tertutup kemungkinan pihak Provider berperan sebagai Agent/Broker asalkan memiliki fasilitas layanan publik yang sudah tersosialisasi dan dikenal oleh publik. Masing-masing peran Sistem Informasi tersebut dapat dipetakan ke dalam model arsitektur Web Services, sebagaimana yang diilustrasikan dalam Gambar 4.1. berikut ini.



Gambar 4.1. Pemetaan Model Web Services untuk pengambilan data penduduk.

Selain kasus yang dilakukan oleh Sistem Informasi Perpajakan terhadap Sistem Informasi Kependudukan, sebaliknya dapat dilakukan oleh Sistem Informasi Kependudukan terhadap Sistem Informasi Perpajakan, untuk kasus pengambilan informasi data penduduk yang sudah memiliki NPWP. Peran yang dipetakan dalam arsitektur Web Services kali ini adalah Sistem Informasi Kependudukan sebagai Requester dan Sistem Informasi Perpajakan sebagai Provider. Sementara Agent/Broker dianggap sama dengan kasus sebelumnya. Khusus untuk Agent/Broker, di Indonesia sampai saat ini belum ada lembaga khusus yang berperan sebagai Agent/Broker Web Services dengan tugas

mengelola sumberdaya informasi pemerintahan. Ilustrasi pemetaan kasus ini terhadap model Web Services terdapat pada Gambar 4.2. berikut ini.

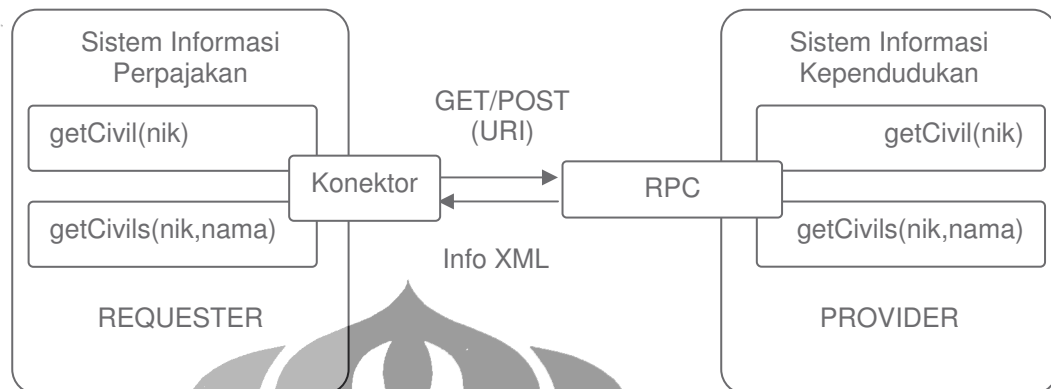


Gambar 4.2. Pemetaan Model Web Services untuk pengambilan data wajib pajak.

Pemetaan studi kasus terhadap model Web Services tersebut perlu dijelaskan dalam mekanisme kerja antar Sistem Informasi melalui metode Web Services. Berdasarkan informasi penelitian mengenai hasil perbandingan metode Web Services yang terdapat pada landasan teori, maka penelitian ini memilih metode REST sebagai metode proses dalam Web Services.

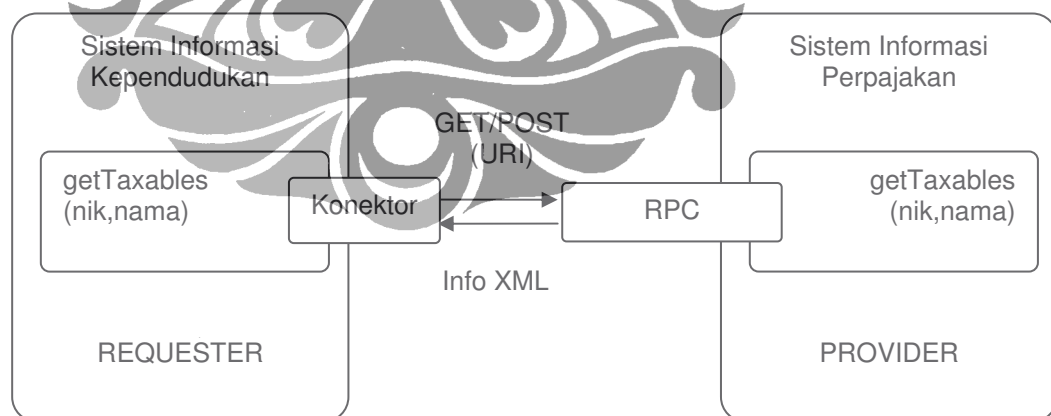
Melalui metode REST, maka pada kasus validasi dan pengambilan data penduduk oleh Sistem Informasi Perpajakan dibedakan menjadi dua bagian, yaitu proses pengambilan seorang data penduduk berdasarkan NIK untuk validasi, dan proses pengambilan sejumlah data penduduk berdasarkan NIK dan Nama penduduk untuk registrasi NPWP secara kolektif. Proses-proses tersebut dapat dilakukan melalui Web Services Sistem Informasi Kependudukan, dengan menyediakan fungsi-fungsi remote untuk pengolahan data berupa fungsi untuk mengambil data penduduk berdasarkan NIK dengan nama *getCivil* dan fungsi untuk mengambil sejumlah data penduduk berdasarkan NIK dan Nama penduduk

dengan nama *getCivils*. Masing-masing proses tersebut diilustrasikan pada Gambar 4.3. berikut ini.



Gambar 4.3. Mekanisme pengambilan data penduduk dengan metode REST.

Sedangkan pada kasus pengambilan data wajib pajak yang sudah terdaftar oleh Sistem Informasi Kependudukan, proses tersebut dapat dilakukan pihak Sistem Informasi Perpajakan dengan menyediakan fungsi untuk mengambil data wajib pajak berdasarkan NIK dan Nama penduduk dengan nama *getTaxables*. Mekanisme ini diilustrasikan dalam Gambar 4.4.



Gambar 4.4. Mekanisme pengambilan data wajib pajak dengan metode REST.

Pada mekanisme proses pertukaran data informasi yang dilakukan melalui Web Services ini menunjukkan bahwa pertukaran data tidak dilakukan langsung antar database masing-masing Sistem Informasi melainkan lewat fungsi yang

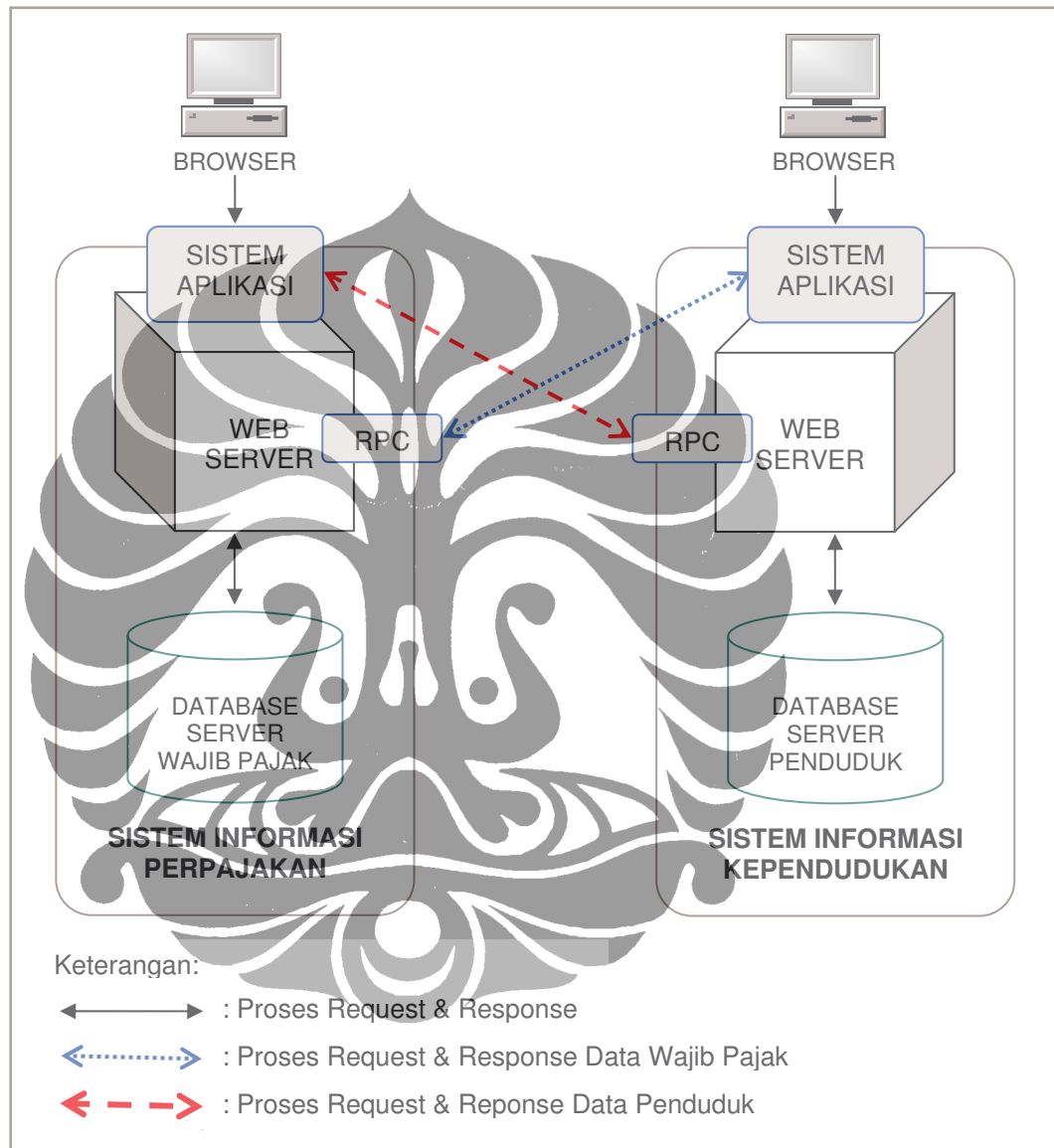
dipublikasikan melalui Web Services. Dengan demikian proses yang dilakukan akan menjadi aman pada tingkat pengolahan data, karena pihak luar tidak bersentuhan langsung dengan database yang ada dalam internal proses suatu Sistem Informasi. Keuntungan lain dengan model seperti ini adalah data dapat diakses tanpa harus tahu platform atau engine driver yang digunakan untuk mengakses suatu database. Hasil dari pemetaan model Web Services dengan metode REST ini menunjukkan adanya orientasi arsitektur yang digunakan, yaitu arsitektur berorientasi pada sumberdaya informasi yang lebih dikenal dengan nama Resource Oriented Architecture (ROA).

Hubungan secara fisik antara Sistem Informasi Perpajakan dan Sistem Informasi Kependudukan melalui Web Services dalam melakukan pertukaran informasi dapat diperjelas melalui Model Infrastruktur Web Services. Model Infrastruktur ini dibuat untuk memahami proses transaksi data antara kedua Sistem Informasi secara online. Ilustrasi sederhana tentang Model Infrastruktur yang digunakan dalam penelitian ini adalah seperti pada Gambar 4.5.

Model Infrastruktur pada Gambar 4.5. menjelaskan bahwa setiap Sistem Informasi yang memiliki sumberdaya informasi publik dapat memberikan akses kepada Sistem Informasi lain yang membutuhkannya, melalui fasilitas RPC atau Web Services yang terdapat pada lokasi Web. Dalam studi kasus ini, kedua Sistem Informasi saling membutuhkan informasi dari keduanya, sehingga memerlukan proses pertukaran data antar Sistem Informasi. Setiap kali Sistem Aplikasi dari domain lain memanggil Web Services, maka masing-masing Web Services atau RPC yang berada pada Web Server akan melakukan pengolahan data ke dalam database melalui fungsi internal yang telah didefinisikan sebelumnya. Fungsi-fungsi tersebut dapat berlokasi di dalam Sistem Aplikasi atau pun di dalam Modul tersendiri, sehingga dapat dipakai secara bersama-sama dengan Sistem Aplikasi yang lain.

Beberapa pendekatan dapat dilakukan pada proses eksekusi fungsi remote (jarak jauh) melalui Web Services atau RPC. Diantaranya adalah proses eksekusi fungsi secara langsung yang dilewatkan melalui parameter URI dengan mencantumkan nama fungsi yang akan diproses. Cara lain adalah dengan proses interpretasi fungsi ke dalam nama umumnya, sehingga pada saat nama umum

fungsi yang dilewatkan melalui parameter URI tidak langsung dieksekusi oleh Web Services akan tetapi diterjemahkan dahulu ke dalam daftar nama fungsi internal. Hal ini dilakukan umumnya untuk menjaga keamanan dari proses pihak luar yang ingin melakukan akses langsung ke dalam fungsi-fungsi internal.



Gambar 4.5. Model Infrastruktur Interoperabilitas Sistem Informasi.

4.2. PERANCANGAN MODEL INTEROPERABILITAS

Perancangan model ini disusun berdasarkan hasil analisa model yang diawali dengan pembuatan skema data dari masing-masing Sistem Informasi yang akan melakukan pertukaran data. Skema data yang perlu dipersiapkan untuk

Sistem Informasi Kependudukan diambil berdasarkan sumber skema data pada KTP, sedangkan skema data yang diperlukan untuk Sistem Informasi Perpajakan bersumber dari Surat Keterangan Terdaftar Wajib Pajak. Ilustrasi pada Tabel 4.2. menggambarkan skema data penduduk yang terdiri dari Nomor Induk Kependudukan (NIK), Nama, Jenis Kelamin, Tempat Lahir Tanggal Lahir, Alamat, Kelurahan Kecamatan, Kota, Agama, Golongan Darah, Kewarganegaraan, Tanggal Pengurusan, Batas Berlaku, Petugas Pencatat dan Waktu Tercatat. Banyaknya karakter yang dapat ditampung dalam NIK adalah 16 (enam belas) sesuai dengan NIK yang berlaku pada sumber data KTP. Beberapa field yang menjadi tambahan peneliti adalah tanggal_urus, pencatat dan tercatat. Peneliti menganggap perlu menambah field tersebut untuk melacak lamanya batas berlaku dan informasi perubahan terhadap data melalui Sistem Informasi Kependudukan.

Tabel 4.1. Skema Data Penduduk

Field	Type	Comments
nik	varchar(16)	Nomor Induk Kependudukan
nama	text	Nama lengkap
jenis	text	Jenis kelamin
tempat_lahir	text	Tempat kelahiran
tanggal_lahir	date	Tanggal lahir
alamat	text	Alamat
kelurahan	text	Kelurahan
kecamatan	text	Kecamatan
kota	text	Kota
agama	text	Agama
darah	text	Golongan Darah
warganegara	text	Kewarganegaraan
tanggal_urus	date	Tanggal pengurusan
batas_berlaku	date	Batas masa berlaku
pencatat	text	Petugas pencatat
tercatat	datetime	Tanggal pencatat

Sementara itu pada Tabel 4.2. menggambarkan skema data wajib pajak sesuai dengan sumber data Surat Keterangan Terdaftar Wajib Pajak.

Tabel 4.2. Skema Data Wajib Pajak

Field	Type	Comments
npwp	varchar(16)	Nomor pokok wajib pajak
nik	text	Nomor Induk Kependudukan
nama	text	Nama lengkap
tempat_lahir	text	Tempat kelahiran
tanggal_lahir	date	Tanggal lahir
alamat	text	Alamat
kelurahan	text	Kelurahan
kecamatan	text	Kecamatan
kota	text	Kota
lapangan_usaha	text	Klasifikasi lapangan usaha
status_modal	text	Modal kepemilikan
nomor_register	text	Nomor terdaftar
tanggal_register	date	Tanggal terdaftar
surat_register	text	Surat keputusan terdaftar
pph4	tinyint(1)	PPh Pasal 4 (2)
pph15	tinyint(1)	PPh Pasal 15
pph19	tinyint(1)	PPh Pasal 19
pph21	tinyint(1)	PPh Pasal 21
pph22	tinyint(1)	PPh Pasal 22
pph23	tinyint(1)	PPh Pasal 23
pph25	tinyint(1)	PPh Pasal 25
pph26	tinyint(1)	PPh Pasal 26
pph29	tinyint(1)	PPh Pasal 29
pencatat	text	Petugas pencatat
tercatat	datetime	Tanggal pencatatan

Field tambahan peneliti yang bukan berasal dari sumber informasi adalah NIK, Tempat Lahir, Tanggal Lahir, Petugas Pencatat dan Waktu Tercatat. Hal ini dilakukan untuk memudahkan pengambilan data penduduk menggunakan NIK dan memastikan pertukaran data berhasil dilakukan. Berdasarkan hasil analisa model, maka didapatkan kaitan antara kedua skema data tersebut.

Dengan demikian perancangan Model Interoperabilitas dapat dimulai dengan melakukan beberapa tahap perancangan berdasarkan entitas model arsitektur Web Service dengan metode REST.

1. Rancangan Penyedia Sumberdaya Informasi (Provider)

Pada rancangan ini terdapat beberapa proses yang perlu dilakukan bagi entitas Provider, diantaranya:

- a. Membuat fungsi yang akan digunakan sebagai sumberdaya informasi publik. Algoritma yang digunakan dalam fungsi ini adalah:
 - 1) Buka koneksi database melalui fungsi pustaka database yang tersedia.
 - 2) Jika koneksi terhubung, ambil data berdasarkan kriteria permintaan dan kembalikan nilai fungsi dengan hasil data array.
 - 3) Jika tidak terkoneksi atau data tidak ditemukan, kembalikan nilai fungsi dengan nilai kesalahan (false).
- b. Menentukan struktur format parameter Uniform Resource Identifier (URI) yang akan dijadikan acuan sebagai deskripsi publikasi Web Services dalam bentuk Web Application Description Language (WADL). Dalam kasus ini parameter yang akan digunakan adalah:
 - *Nama resource (resource)*: menunjukkan nama sumberdaya informasi yang akan dipetakan ke dalam fungsi yang sudah didefinisikan sebelumnya.
 - *Format data masukan (inputformat)*: menunjukkan jenis format data masukan yang dilewatkan ke dalam Web Services atau RPC, berupa format teks biasa (txt) atau xml.
 - *Format hasil (outputformat)*: menunjukkan jenis format hasil keluaran atau nilai kembalian fungsi pengambilan data, berupa format teks biasa (txt), xml atau html.

- *Data masukan (inputdata)*: menunjukkan nilai atau kriteria permintaan sebagai data masukan yang dilewatkan ke dalam fungsi melalui RPC. Misalnya nama=AGUS atau kurs=RUPIAH. Jika format masukannya adalah teks biasa maka sintaksnya adalah:

nama_variabel= nilai[;nama_variabel=nilai]

Pemisah antar data diberi tanda ”;”. Contohnya:

inputdata=nama=AGUS;lahir=21/04/1970

Sebaliknya bila format masukannya adalah XML maka sintaksnya adalah:

<request>
<nama_variabel>nilai</nama_variabel>
[<nama_variabel>nilai</nama_variabel>]
</request>

Contohnya:

inputdata=<request ><nama>AGUS</nama></request>

Secara keseluruhan struktur format WADL yang digunakan dalam URI dapat dicontohkan sebagai berikut:

http://siduk.net/rpc/?resource=funct&inputformat=txt&outputformat=xml&inputdata=nama=AGUS

atau

http://siduk.net/rpc/?resource=funct
&inputformat=txt&outputformat=xml
&inputdata=< request ><nama>AGUS</nama></request >

- Membuat script program aplikasi Web Services yang berfungsi sebagai RPC, untuk mengolah request parameter URI (WADL) yang dikirim oleh entitas Requester, agar dapat memvalidasi dan memberikan data yang

diminta melalui pemetaan parameter resource terhadap nama fungsi internal. Pemetaan nama resource terhadap nama fungsi dilakukan karena disamping untuk keamanan interaksi dari pihak luar, nama fungsi biasanya memiliki format yang lebih spesifik pada bahasa pemrograman, sedangkan nama resource memiliki format secara umum yang dapat dimengerti oleh publik. Contohnya nama fungsi getCivil, memiliki nama resource Penduduk. Algoritma yang dapat dibuat untuk program Web Services ini adalah:

- 1) Menentukan absolute path berkas Web Services sebagai basis path.
- 2) Menggabungkan modul-modul fungsi terkait ke dalam program.
- 3) Memeriksa kelengkapan struktur parameter URI yang sudah didefinisikan sebagai standar parameter dalam pemanggilan Web Services.
- 4) Jika parameternya lengkap, maka petakan nama resource ke dalam nama fungsi, konversi format data masukan ke dalam variabel untuk parameter fungsi, jika fungsi ada jalankan fungsi dengan masukan parameter, setelah fungsi berhasil diproses kembalikan nilai hasil fungsi ke dalam variabel hasil, jika permintaan format hasil berupa XML atau HTML maka konversikan nilai variabel hasil.
- 5) Jika parameter tidak lengkap, nama resource tidak dapat dipetakan ke dalam fungsi, fungsi tidak ditemukan, atau tidak ada hasil proses fungsi, maka isi nilai variabel hasil dengan pesan kesalahan atau spasi kosong.
- 6) Akhiri proses dengan cara mencetak nilai variabel hasil sebagai script yang akan diolah oleh Web Server.

2. Rancangan Pendaftaran dan Penemuan Sumberdaya (Registry/Discovery)

Pada bagian pendaftaran dan penemuan layanan, perlu dibuat beberapa pencatatan dalam database agar dokumentasinya mudah untuk dikelola dan informasinya mudah untuk dicari. Beberapa langkah yang harus dipersiapkan

dalam pembuatan aplikasi pendaftaran layanan (sumberdaya informasi), adalah sebagai berikut:

- a. Membuat struktur database yang akan digunakan dalam pencatatan dan penemuan informasi layanan publik. Bentuk skema datanya diilustrasikan pada Tabel 4.3.
- b. Membuat aplikasi web server yang digunakan untuk registrasi Provider beserta layanannya.
- c. Membuat aplikasi web server yang dapat diakses oleh publik dan menyediakan modul untuk registrasi bagi pihak Requestor dalam mendapatkan kunci akses publik.

Tabel 4.3. Skema Data Sumberdaya Informasi (Resource)

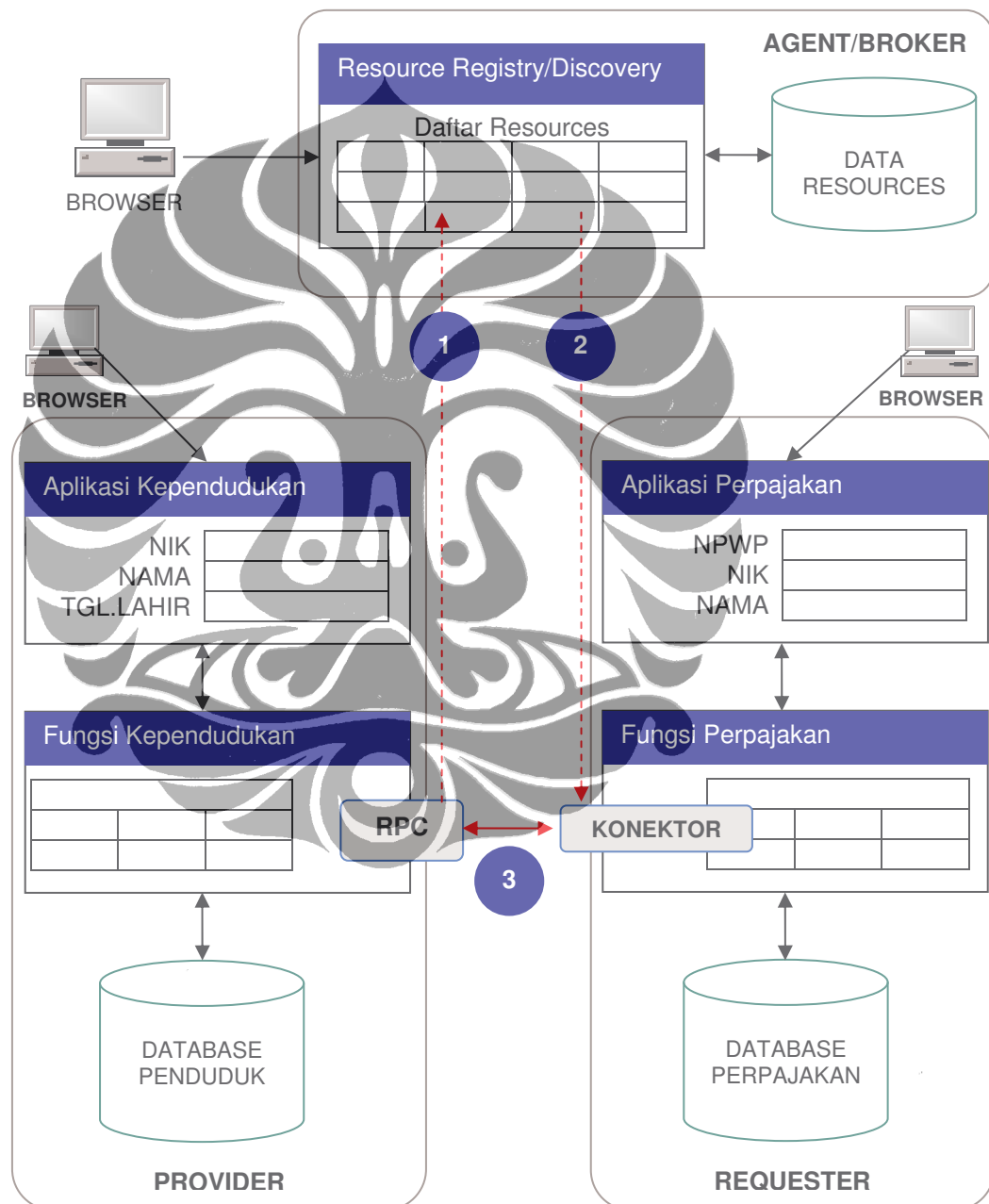
Field	Type	Comments
id	bigint(20)	Nomor identitas
profil	text	Identitas provider
nama	text	Nama sumberdaya
alamat	longtext	Alamat sumberdaya (URL)
format_masukan	text	Format masukan (XML/TEXT)
data_masukan	longtext	Data masukan
format_keluaran	text	Format keluaran (XML/HTML/TEXT)
keterangan	text	Keterangan
publikasi	text	Status publikasi
pencatat	text	Pengelola sumberdaya
tercatat	text	Waktu perekaman

3. Rancangan Pengguna (Requester)

Pihak pengguna dapat langsung menggunakan fungsi layanan setelah mendapatkan baris perintah URL pada browser yang tersedia di lokasi client. Namun bila hasil dari fungsi layanan tersebut akan digunakan sebagai sumber data olahan, maka baris perintah tersebut perlu dimasukkan ke dalam kode program pengguna melalui fungsi komunikasi yang tersedia dalam pustaka atau modul bahasa pemrograman. Komunikasi melalui kode program

umumnya menggunakan fungsi komunikasi socket atau pengolahan file jarak jauh yang dimiliki oleh beberapa pemrograman aplikasi web server, diantaranya PHP, ASP, atau JSP.

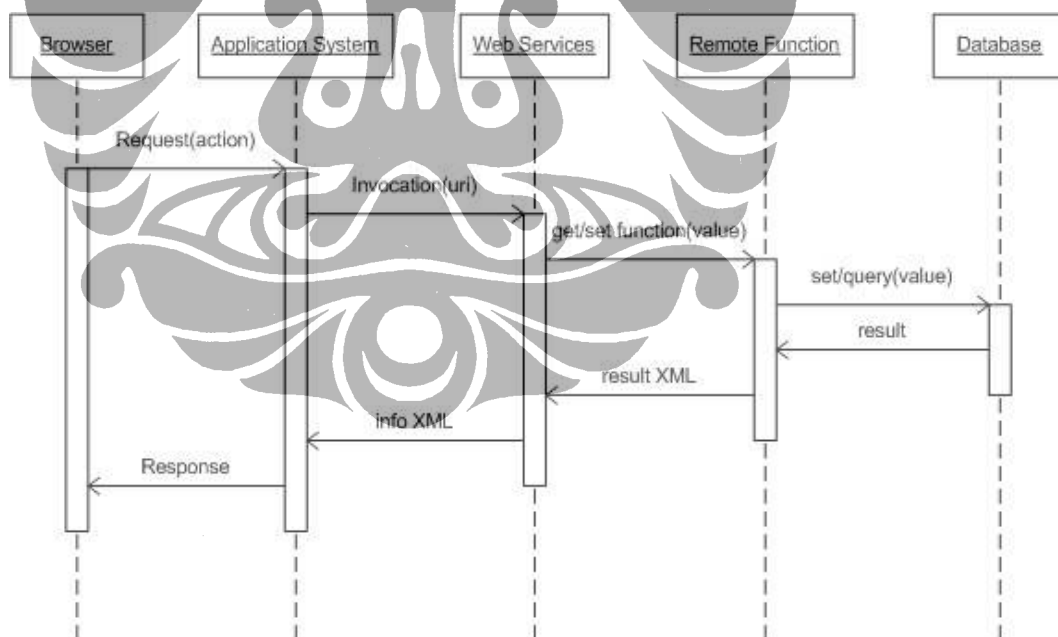
Tahapan proses perancangan tersebut dapat mempengaruhi langkah-langkah implementasi Interoperabilitas antar Sistem Informasi, seperti diilustrasikan pada Gambar 4.6. berikut ini.



Gambar 4.6. Implementasi Model Interoperabilitas Sistem Informasi

Implementasi Model Interoperabilitas Sistem Informasi diawali dengan proses publikasi sumberdaya informasi kependudukan melalui pendaftaran sumberdaya informasi serta memberi keterangan akses informasi ke dalam sistem aplikasi berbasis web yang dikelola oleh Agent/Broker (Gambar 4.6. Tahap 1). Melalui publikasi ini pihak Requester akan mendapatkan keterangan akses informasi untuk diunduh kedalam kode program pemanggilnya melalui fungsi koneksi PHP-CURL (Gambar 4.6. Tahap 2). Selanjutnya pihak Requester dapat memperoleh data berdasarkan kriteria yang diinginkan (Gambar 4.6. Tahap 3).

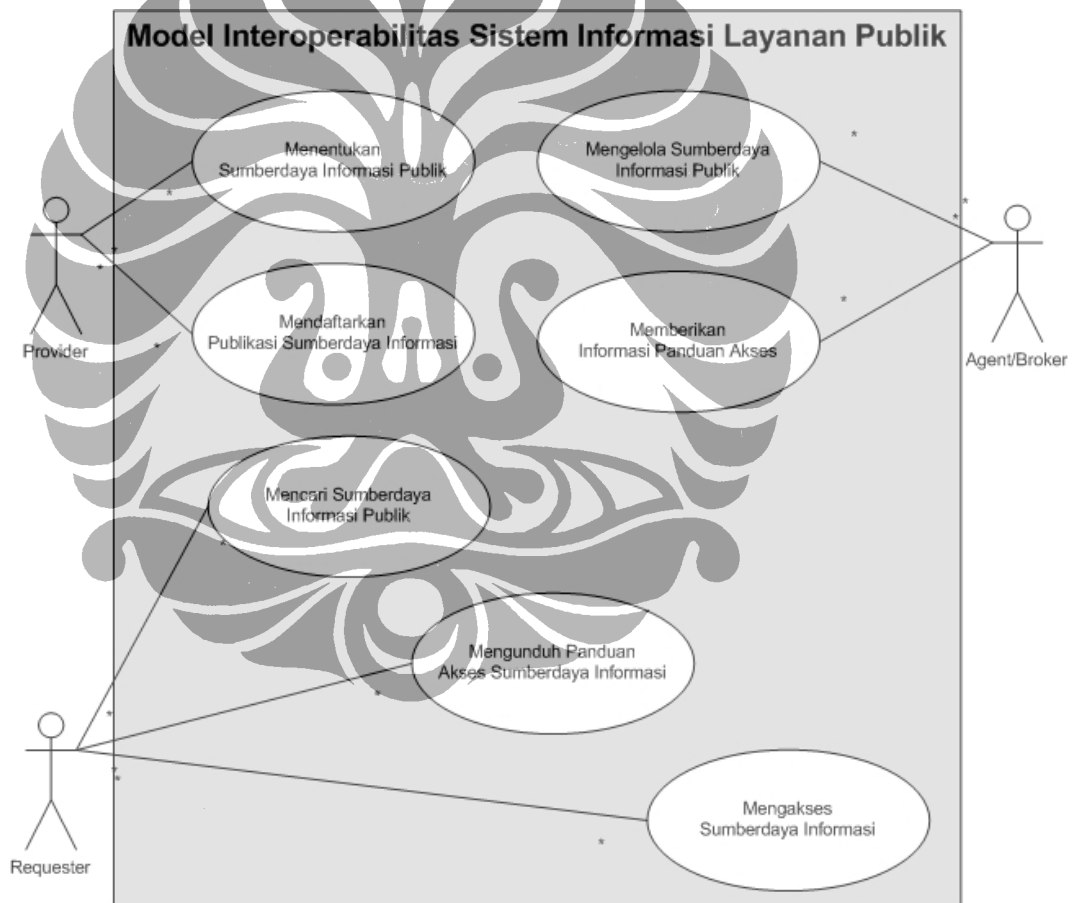
Berdasarkan mekanisme transaksi pertukaran informasi yang telah dijelaskan pada tahap analisa, rancangan implementasi dari metode REST tersebut dapat diinterpretasikan ke dalam diagram Sequence UML. Dengan demikian proses yang dilakukan selama perjalanan pengambilan data melalui Web Services atau RPC dapat tergambar dengan jelas bagaimana suatu data diperoleh. Ilustrasi yang diberikan dalam diagram ini merupakan pandangan secara umum tentang proses pengambilan data melalui Web Services. Berikut adalah gambarannya.



Gambar 4.7. Diagram Sequence Model Interoperabilitas

Proses yang terjadi dalam Gambar 4.7. diasumsikan untuk pengambilan data dari tempat yang terpisah jauh dari Sistem Aplikasi. Apabila sebuah browser mengirimkan permintaan terhadap Sistem Aplikasi Web berupa proses

pengambilan data, maka setelah permintaan tersebut diterima, sistem aplikasi akan melakukan pemanggilan fungsi melalui fasilitas konektor dan meneruskan nilai permintaan yang diminta oleh browser. Sesampainya nilai/pesan permintaan tersebut di dalam Web Services/RPC, maka Web Services akan memanggil dan memproses fungsi yang tersedia dalam modul Remote Function. Fungsi tersebut kemudian akan melakukan proses pengolahan data ke dalam database berdasarkan nilai yang diterima, setelah selesai nilai balik akan dikembalikan kepada Web Services dalam format teks atau XML. Data XML tersebut selanjutnya diverifikasi dan dikembalikan kepada Sistem Aplikasi pemanggilnya. Lalu Sistem Aplikasi menyajikan informasi tersebut dalam format HTML kepada Browser.

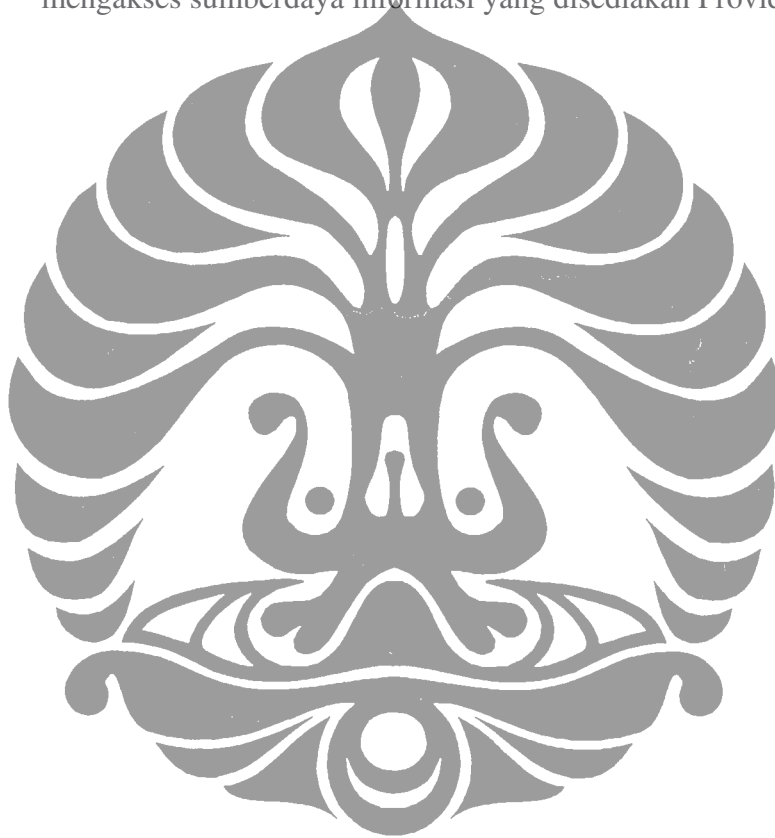


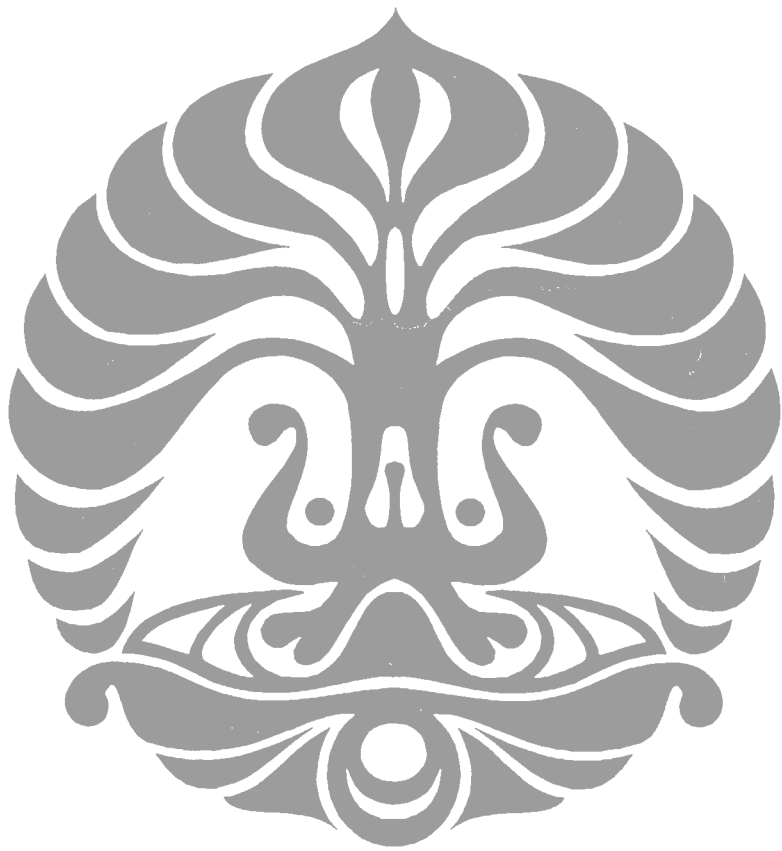
Gambar 4.8. Diagram Use Case Model Interoperabilitas.

Secara garis besar langkah-langkah yang dilakukan dalam proses penerapan Interoperabilitas antar Sistem Informasi tersebut dapat dijadikan Model Interoperabilitas Sistem Informasi Layanan Publik dalam bentuk diagram Use-

Case UML pada Gambar 4.7. Dimana masing-masing pihak atau entitas memiliki peran sebagai berikut:

1. Provider memiliki peran dalam menentukan sumberdaya informasi publik dan mendaftarkan publikasi sumberdaya informasi.
2. Agent/Broker memiliki peran mengelola website sumberdaya informasi publik dan memberikan informasi panduan akses.
3. Requester memiliki peran mencari sumberdaya informasi publik melalui website Agent/Broker, mengunduh panduan akses sumberdaya informasi, dan mengakses sumberdaya informasi yang disediakan Provider.





BAB V

IMPLEMENTASI DAN UJI COBA

5.1. IMPLEMENTASI

Hal-hal yang perlu dipersiapkan dalam proses implementasi penelitian ini adalah sebagai berikut.

1. Mempersiapkan perangkat keras yang akan digunakan, minimal memiliki spesifikasi seperti pada tabel berikut ini.

Tabel 5.1. Spesifikasi Perangkat Keras

<i>Kebutuhan</i>	<i>Spesifikasi</i>
Processor	Intel Pentium 4 (minimum)
RAM	256 MB
Harddisk	160 GB
VGA Memory	64 MB
LAN	UTP 10/100
Monitor VGA	1024x768 True Color
Keyboard	101-key (minimum)
Mouse	Standard

2. Mempersiapkan perangkat lunak berupa aplikasi yang digunakan untuk pembuatan aplikasi berbasis web berdasarkan spesifikasi pada tabel berikut.

Tabel 5.2. Spesifikasi Perangkat Lunak

<i>Kebutuhan</i>	<i>Spesifikasi</i>
Sistem Operasi	GNU-Linux dan Microsoft Windows
Web Server	Apache (LAMP/PHP)
Web Programming Language	PHP, HTML, Javascript
RDBMS	MySQL dan PHPMyAdmin
Pustaka (Library)	Client-URL, PHP Data Object, dan SimpleXML

3. Mempersiapkan jaringan internet dengan akses protokol http melalui 3 (tiga) alamat internet protokol (IP Address) pada bandwidth 1Mbps.

Tabel 5.3. Daftar IP Address

<i>Kebutuhan</i>	<i>Spesifikasi</i>
Sistem Informasi Kependudukan	127.0.0.1 atau 202.153.132.126
Agent/Broker	127.0.0.1 atau 202.153.132.116
Sistem Informasi Perpajakan	127.0.0.1 atau 202.153.132.116

Dengan asumsi bahwa seluruh kebutuhan persiapan sudah terpasang dan dapat digunakan sebagaimana mestinya, maka selanjutnya dilakukan proses implementasi melalui langkah-langkah berikut sesuai analisa perancangan:

1. Menentukan fungsi publik pada entitas Provider.

Sesuai dengan kebutuhan pertukaran informasi antar Sistem Informasi dalam Model Interoperabilitas pada kasus ini, maka terdapat beberapa fungsi publik yang diperlukan oleh masing-masing Sistem Informasi tersebut. Fungsi publik yang perlu disediakan oleh Sistem Informasi Kependudukan adalah fungsi pengambilan data penduduk berdasarkan kriteria NIK dengan nama *getCivil(nik)* dan fungsi pengambilan sejumlah data penduduk berdasarkan sebagian kriteria NIK atau Nama penduduk dengan nama *getCivilList(nik,nama)*. Sementara itu fungsi publik yang disediakan oleh Sistem Informasi Perpajakan adalah fungsi pengambilan sejumlah data wajib pajak berdasarkan sebagian kriteria NIK dan Nama wajib pajak dengan nama *getTaxableList(nik,nama)*.

Bentuk penamaan fungsi kependudukan meliputi *getCivil*, *getCivilXML*, *getCivilList* dan *getCivilListXML*, serta penamaan fungsi wajib pajak meliputi *getTaxableList*, dan *getTaxableListXML*. Penamaan XML pada akhiran nama fungsi digunakan sebagai penunjuk bahwa data hasil kembalian fungsi tersebut dalam bentuk format teks XML. Umumnya dalam Web Services fungsi yang menghasilkan data format XML digunakan sebagai remote function atau fungsi publik. Kode program yang dibuat pada masing-masing fungsi, didasari oleh algoritma pengambilan data yang telah dibahas pada perancangan model dalam penelitian ini. Dengan demikian implementasi algoritma berupa kode program

PHP untuk fungsi pengambilan data penduduk dapat dicontohkan dalam daftar Program 5.1.

**Program 5.1. Fungsi pengambilan data penduduk berdasarkan kriteria
Nomor Induk Kependudukan**

```
function getCivil($id) {
try {
    $dbh = new PDO(DB_SOURCE, DB_USER, DB_PASSWORD);
    $sth = $dbh->prepare("SELECT * FROM `penduduk` ".
        "WHERE `nik` = :id;");
    $sth->bindParam(":id", $id);
    $sth->execute();
    $index=0;
    foreach($sth->fetchAll() as $rset) {
        $result=array("id"=>$rset['nik'],
            "name"=>$rset['nama'],
            "sex"=>$rset['jenis'],
            "born"=>$rset['tempat_lahir'],
            "birth"=>$rset['tanggal_lahir'],
            "address"=>$rset['alamat'],
            "district"=>$rset['kelurahan'],
            "town"=>$rset['kecamatan'],
            "city"=>$rset['kota'],
            "religion"=>$rset['agama'],
            "blood"=>$rset['darah'],
            "citizen"=>$rset['warganegara'],
            "expired"=>$rset['batas_berlaku'],
            "administrator"=>$rset['pencatat'],
            "updated"=>$rset['tercatat']);
    }
    $dbh = null;
}
catch (PDOException $e) {
    file_put_contents(ERRLOG, $e->getMessage());
    die('Error: '.$e->getMessage());
}
return $result;
}
```

Program 5.2. Fungsi publik pengambilan data penduduk berdasarkan kriteria Nomor Induk Kependudukan

```
function getCivilXML($id) {
    $result=getCivil($id);
    if($result){
        $items=$result;
        $result="<valid_response>";
        foreach($items as $key=>$val){
            $result.="<".$key.">".$val."</".$key.">";
        }
        $result.="</valid_response>";
    }
    return $result;
}
```

Pada Program 5.1. dijelaskan bahwa fungsi tersebut akan menghasilkan data penduduk dalam format array jika berhasil dan boolean false kalau tidak berhasil. Sedangkan fungsi pada Program 5.2. merupakan fungsi konversi hasil fungsi pada program 5.1. sehingga akan mengembalikan data penduduk dalam format XML jika berhasil.

2. Mendefinisikan nama sumberdaya publik pada Web Services.

Nama sumberdaya publik atau umumnya dalam penelitian ini disebut resource, perlu didefinisikan dalam kode program Web Services atau RPC untuk proses pemetaan terhadap nama fungsi publik yang telah ditentukan dalam modul fungsi tertentu. Web Services ini akan menerima setiap permintaan berdasarkan informasi dari URI. Agar parameter URI tersebut dapat dikirim melalui metode GET maupun POST, maka proses pemeriksaan parameter dilakukan melalui variabel global REQUEST yang disediakan oleh PHP. Melalui variabel global REQUEST ini, komposisi parameter yang dikirimkan dapat diketahui kelengkapannya. Apabila salah satu nama parameter tidak disebutkan maka pengecekan terhadap array dalam variabel REQUEST tersebut tidak menemukannya. Sehingga proses selanjutnya tidak akan dilakukan, dengan kata

lain Web Services akan mengembalikan data kosong. Untuk membedakan data hasil proses yang berhasil dengan data yang bukan merupakan hasil proses fungsi dalam Web Services, setiap data yang valid diberi inisial "REST:" pada awal isi data yang akan dikembalikan kepada fungsi interface Requester. Contoh kode program Web Services tersebut dijabarkan dalam kode Program 5.3.

Program 5.3. Program Web Services atau RPC

```

<?php
/** ---Menentukan ABSPATH sebagai path direktori **/
if ( !defined('ABSPATH') )
    define( 'ABSPATH', dirname(__FILE__) . '/' );
/** --Sisipan tambahan untuk modul fungsi **/
require_once(ABSPATH.'includes/constant.inc.php');
require_once(ABSPATH.FUNCTION_DIR.'/pengkodean.php');
require_once(ABSPATH.FUNCTION_DIR.'/penduduk.php');
/** Akhir sisipan -----**/
if(isset($_REQUEST["resource"]) && isset($_REQUEST["inputformat"])
&& isset($_REQUEST["outputformat"])
&& isset($_REQUEST["parameter"])){
    $result=parseResource($_REQUEST["resource"],
    $_REQUEST["inputformat"],$_REQUEST["outputformat"],
    $_REQUEST["parameter"]);
    echo $result;
} /*--- end of operation ---*/
function getFunction($resource){
    switch ($resource) {
/** - Sisipan tambahan untuk pemetaan resource terhadap fungsi**/
        case "seorang-penduduk":
            $result="getCivilXML";break;
        case "beberapa-penduduk":
            $result="getCivilListXML";break;
        case "lembaran-penduduk":
            $result="getCivilPageXML";break;
/** Akhir sisipan -----**/
        default:$result="";}
    return $result;
}

```

(Sambungan)

```

/*----- Fix parser routine function -----*/
function parseResource($resource,$inputStyle,$outputStyle,$input){
    $onfunction=getFunction($resource); $aParam=array();
    if($inputStyle=="xml"){
        /* Convert XML parameter format into array variable */
        $aParam=parseXML2Array($input);
    }
    else {
        /* Convert TXT parameter format into array variable */
        $aParam=parseTXT2Array($input);
    }
    $result="";
    if(function_exists($onfunction) && $aParam){
        /* Do in exist function and parameter */
        $result=call_user_func($onfunction, $aParam);
    }
    else{
        if($outputStyle=="xml"){
            $result="<invalid_response>Pemanggilan resource ".
            $resource." tidak valid.</invalid_response>";
        }
        else {
            $result="Pemanggilan resource ".$resource.
            " tidak valid.";
        }
    }
    if($result!=""){
        switch($outputStyle){
            case "html":
                $result="REST:".parse2HTML($result);break;
            case "xml":
                $result="REST:".parse2XML($result); break;
            case "txt":
                $result="REST:".$result;break;
        }
    } /*--- end of convert ---*/
    return $result;
}

```

(Sambungan)

```

function parseXML2Array($param) {
    $aParam=array();
    $xml = new SimpleXMLElement($param);
    $rootName=$xml->getName();
    if (strtolower($rootName)=="request") {
        foreach($xml->children() as $items){
            $aParam[$items->getName()]=$items;
        }
    }
    return $aParam;
}

function parseTXT2Array($param) {
    if(strpos($param,";")==false){
        list($key,$val)=explode("=", $param);
        $aParam[$key]=$val;
    }
    else {
        $items=explode(";", $param);
        foreach($items as $item):
            list($key,$val)=explode("=", $item);
            $aParam[$key]=$val;
        endforeach;
    }
    return $aParam;
}

function parse2HTML($param) {
    $result=$param;
    return $result;
}

function parse2XML($param) {
    $xml = new SimpleXMLElement($param);
    $result=$xml->asXML();
    return $result;
}
?>

```

Dalam kode program Web Services atau RPC tersebut disediakan ruang untuk penyisipan kode program berupa pemuatan modul fungsi dan pemetaan nama resource terhadap nama fungsi. Pada contoh kasus data kependudukan Web Services memberikan nama resource *seorang-penduduk* untuk dipetakan terhadap fungsi yang bernama *getCivilXML*, kemudian nama resource *beberapa-penduduk* dipetakan ke dalam fungsi *getCivilListXML*. Sebaliknya pada Web Services wajib pajak disediakan nama resource *beberapa-wajibpajak* yang dipetakan ke dalam fungsi *getTaxableListXML*. Dengan demikian fungsi-fungsi tersebut dapat diproses tanpa diketahui secara fisik oleh Requester. Hasil pengolahan data akan dikonversi melalui fungsi-fungsi parser sesuai dengan format keluaran yang didefinisikan sebelumnya lewat parameter URI.

3. Menyediakan aplikasi berbasis web untuk proses registrasi dan penemuan sumberdaya informasi (Registry/Discovery) pada entitas Agent/Broker.

Berdasarkan kepentingan pengguna maka Sistem Aplikasi Resource Registry dan Discovery dipisah menjadi dua bagian. Pertama untuk proses registrasi bernama Resource Registry dan kedua untuk proses pencarian sumberdaya informasi diberi nama Resource Discovery. Fasilitas yang terdapat dalam Aplikasi Resource Registry adalah berupa modul pengelolaan daftar katalog sumberdaya informasi yang dapat diatur oleh pihak penyedia (Provider). Bentuk fasilitas Aplikasi Resource Registry diilustrasikan pada Gambar 5.1.

No.	Penyedia	Nama Resource	Alamat (URL)	Format Masukan	Format Hasil	Keterangan	Publikasi	Pendata	Terdata	Ralat
1	depdiknas	jumlah	http://202.153.132.126/diknas	txt	txt	Fungsi operasi penjumlahan	<input checked="" type="checkbox"/>	budi	2009-07-07 00:24:08	
2	depdiknas	pembagian	http://202.153.132.126/diknas	txt	txt	Fungsi operasi pembagian	<input checked="" type="checkbox"/>	budi	2009-07-07 00:21:45	
3	depdiknas	perkalian	http://202.153.132.126/diknas/	txt	txt	Fungsi operasi perkalian	<input type="checkbox"/>	budi	2009-07-07 00:21:57	

Gambar 5.1. Layar Daftar Resource pada Aplikasi Registry

Sementara itu fasilitas yang diberikan pada Aplikasi Resource Discovery adalah daftar pencarian atau penemuan resource, deskripsi resource yang dapat diunduh, dan uji coba Web Services, seperti yang terlihat dalam Gambar 5.2. Database Management System untuk pendataan informasi resource ini menggunakan MySQL, skema data tabel yang digunakan adalah berdasarkan hasil rancangan sesuai kebutuhan penyimpanan informasi tentang Web Services.



The screenshot shows a web browser window with the URL <http://202.153.132.116/wsdiscovery/>. The page title is "Resources Discovery". There are search fields for "Penyedia" and "Nama Resource" with a "Cari" button. Below the search fields are pagination controls showing "Halaman: << < 1 > >>" and "Total: 1 Halaman (5 Baris) atau 5 Records". A table displays the following data:

No.	Penyedia	Nama Resource	Format Masukan	Format Hasil	Tercatat	Keterangan	Coba
1	depdagri	beberapa-penduduk	txt	xml	2009-04-15 00:26:58		
2	depdagri	lembaran-penduduk	txt	xml	2009-04-15 00:33:18		
3	depdagri	seorang-penduduk	txt	xml	2009-04-15 00:14:09		
4	depkeu	beberapa-wajibpajak	txt	xml	2009-04-15 00:40:55		
5	depkeu	lembaran-wajibpajak	txt	xml	2009-04-15 00:44:38		

Gambar 5.2. Layar Daftar Resource pada Aplikasi Discovery

Setiap informasi resource yang terdaftar dalam Agent/Broker dapat diakses oleh publik, terutama pihak Requester yang akan menggunakan deskripsi resource untuk melakukan proses pengolahan data yang disediakan pihak Provider. Deskripsi informasi resource tersebut diilustrasikan pada Gambar 5.3.



The screenshot shows the "Rincian Resource" page. The details are as follows:

- Nama Resource:** seorang-penduduk
- Penyedia:** depdagri
- Alamat (URL):** <http://202.153.132.116/siduk/pc.php>
- Keterangan:** Nilai kembalian jika valid

```

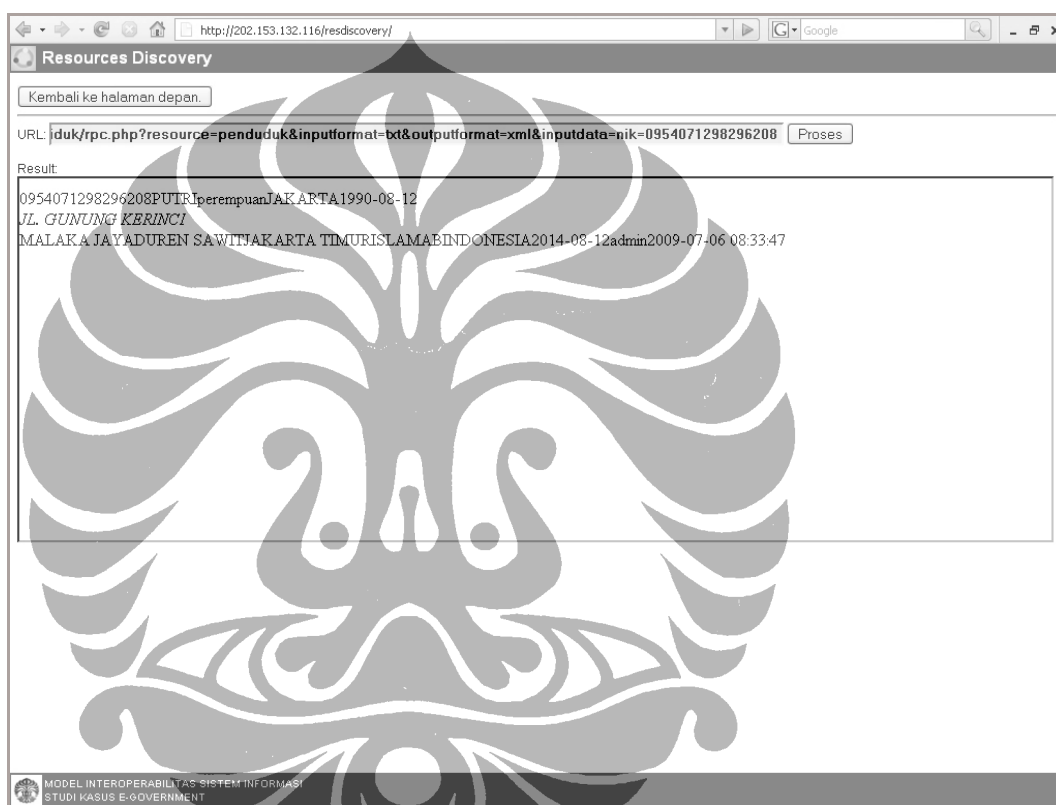
REST:<?xml version="1.0">
<valid response>
  <id></id>
  <name></name>
  <sex></sex>
  <born></born>
  <birth></birth>
  <address></address>
  <district></district>
  <town></town>

```
- Format Masukan:** txt
- Data Masukan:** nik=1234567890123456
- Format Keluaran/Hasil:** xml

A "Kembali" button is located at the bottom right of the form.

Gambar 5.3. Layar Deskripsi Resource pada Aplikasi Discovery

Fasilitas uji coba Web Services pada Aplikasi Resource Discovery merupakan fasilitas yang paling menguntungkan pihak Requester, karena dengan fasilitas ini dapat diketahui sampai berapa jauh tingkat kebutuhan Requester terhadap resource yang diinginkannya. Sehingga melalui baris perintah dalam URL yang dipublikasikan tersebut dapat diunduh ke dalam kode program yang akan dijadikan invocator fungsi oleh pihak Requester. Bentuk layar uji coba tersebut diilustrasikan pada Gambar 5.4.



Gambar 5.4. Layar Uji Coba Web Services pada Aplikasi Discovery.

4. Mengunduh URI ke dalam Sistem Aplikasi milik entitas Requester

Dengan cara mengunduh URI dari Resources Discovery dan membuat fungsi interface untuk pengambilan sumberdaya informasi pada Sistem Aplikasi Requester maka pengambilan data berdasarkan kriteria tertentu dapat dilakukan melalui pengisian kriteria yang diinginkan pada parameter URI dan mengeksekusi fungsi konektor sebagai penghubung komunikasi sistem berbasis Web.

Proses interkoneksi tersebut dapat dilakukan dengan menggunakan fasilitas pustaka PHP-CURL. Sesampainya permintaan ke dalam program Web

Services atau program RPC yang berada pada sistem informasi pihak Provider maka permintaan tersebut akan diteruskan ke dalam fungsi sampai menghasilkan informasi yang dikembalikan kepada pihak Requester. Contoh kode program konektor pada penelitian ini dibuat dalam fungsi *getRPC(url,parameter)* pada daftar Program 5.4.

Program 5.4. Program Konektor Web Services atau RPC.

```

<?php
function getRPC($url,$param){
    $ch = curl_init();
    curl_setopt($ch, CURLOPT_URL, $url);
    curl_setopt($ch, CURLOPT_POST, TRUE);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, TRUE);
    curl_setopt($ch, CURLOPT_POSTFIELDS, $param);
    $result=curl_exec($ch);
    file_put_contents(ERRLOG, "RESULT:". $result);
    if ($error = curl_error($ch)) {
        $result = false;
    }
    return $result;
}
?>

```

Dalam kasus proses validasi dan pengambilan data penduduk berdasarkan kriteria NIK yang dilakukan Sistem Informasi Perpajakan, maka deskripsi URI yang disediakan melalui publikasi resource pada website Resource Discovery dapat diunduh ke dalam fungsi atau kode program aplikasi Sistem Informasi Perpajakan. Fungsi tersebut merupakan fasilitas yang terdapat dalam Sistem Informasi Perpajakan yang digunakan untuk melakukan validasi dan pengambilan data penduduk berdasarkan NIK ke dalam layar registrasi wajib pajak baru. Dalam fungsi tersebut terdapat proses konversi format data XML yang diterima dari Web Services ke dalam data array agar proses pengolahan data dapat dilakukan melalui logika pemrograman. Dengan demikian proses otomatisasi entri data yang dilakukan dalam layar registrasi wajib pajak pada Sistem Informasi

Perpajakan dapat terpenuhi sesuai harapan. Adapun kode program fungsi interface untuk validasi dan pengambilan data penduduk tersebut dapat dijabarkan pada daftar Program 5.5.

Program 5.5. Program fungsi interface pengambilan data penduduk berdasarkan Nomor Induk Kependudukan

```

<?php
function getCivil($id=""){ // Fungsi interface Web Services (RPC)
    $result=false;
    $url="http://202.153.132.126/siduk/rpc.php";
    $request="resource=seorang-penduduk".
        "&inputformat=txt&outputformat=xml".
        "&inputdata=nik=$id";
    $result=getRPC($url,$request);
    if(substr($result,0,5)=="REST:"){ // validasi konten RPC
        $result=substr_replace($result,"",0,5);
    }
    else{ $result=""; }
    if($result){ // validasi konten XML
        if(strtolower(substr($result,0,5))=="<?xml"){
            $xml = new SimpleXMLElement($result);
            if($xml->getName()=="invalid_response"){
                $result=false;
            }
            elseif ($xml->getName()=="valid_response") {
                $result=false;
                foreach($xml->children() as $items){
                    $result[$items->getName()]=$items;
                }
            }
            else { $result=false; }
        }
    }
    return $result;
}
?>

```

5.2. PENGUJIAN MODEL

Pengujian model interoperabilitas sistem informasi layanan publik dilakukan melalui tiga tahap yaitu pengujian registrasi publikasi sumberdaya informasi, pengujian penemuan sumberdaya informasi, dan pengujian pertukaran informasi antar Sistem Informasi. Berikut ini adalah tahap-tahap pengujian tersebut.

1. Pengujian registrasi publikasi sumberdaya informasi

Hal ini ditujukan pada pengujian aplikasi Resources Registry, yang diawali dengan proses registrasi Provider pada menu registrasi di layar otentikasi. Data yang perlu diisi dalam form registrasi tersebut adalah ID, Profil, Nama, dan e-Mail, khusus untuk profil diisi nama organisasi dalam bentuk inisial atau singkatan dan nama profil ini digunakan sepanjang Provider mendaftarkan sumberdaya informasinya. Jadi meskipun ID berbeda dimiliki oleh penggunanya, namun nama profil harus tetap sama jika berada pada satu organisasi atau lembaga. Bentuk tampilan layarnya adalah seperti Gambar 5.5.

Penyedia Baru	
ID	yusuf
Profil	depdagri
Kunci	
Nama	YUSUF
e-Mail	yusuf@gmail.com
Keterangan	
Pasif	<input type="checkbox"/>
<input type="button" value="Simpan"/> <input type="button" value="Batal"/>	

MODEL INTEROPERABILITAS SISTEM INFORMASI
STUDI KASUS E-GOVERNMENT

Gambar 5.5. Layar registrasi Provider baru untuk pihak kependudukan.

Resources Registry Registrasi

Penyedia Baru

ID	sutarno
Profil	depkeu
Kunci	
Nama	SUTARNO
e-Mail	sutarno@yahoo.com
Keterangan	Staf Dit. Jend. Pajak
Pasif	<input type="checkbox"/>

MODEL INTEROPERABILITAS SISTEM INFORMASI
STUDI KASUS E-GOVERNMENT

Gambar 5.6. Layar registrasi Provider baru untuk pihak perpajakan.

Setelah registrasi provider berhasil disimpan, langkah selanjutnya adalah melakukan proses otentikasi untuk masuk ke dalam sistem. Dalam layar otentikasi terdapat pilihan peran yang menunjukkan status pengguna, bisa sebagai pengelola atau penyedia.

Resources Registry Registrasi

Otentikasi

Peran: Penyedia

Pengguna:

Sandi:

Gambar 5.7. Layar otentikasi sebagai Provider

Untuk masuk sebagai Provider maka peran yang dipilih adalah Penyedia, lalu pengisian selanjutnya adalah pengguna dengan ID yang telah diisikan dalam registrasi dan sandi jika ada. Setelah proses otentikasi berhasil diterima sistem, maka tampil menu pilihan Serambi dan Resources. Dengan memilih menu Resources, maka tampil daftar sumberdaya informasi dan tombol navigasi untuk menambah, menghapus, mengaktifkan dan mempasifkan publikasi sumberdaya informasi.

Penambahan sumberdaya informasi yang akan dipublikasikan dilakukan langsung dengan menekan tombol Tambah, akibatnya akan tampil form pengisian pendaftaran sumberdaya informasi seperti yang terlihat pada Gambar 5.8.

The screenshot shows a web browser window with the URL `http://202.153.132.116/resregistry/`. The page title is "Resources Registry" and the user is logged in as "depdagri". The main content area displays a form titled "Resource Baru" with the following fields:

- Nama Resource: seorang-penduduk
- Penyedia: depdagri
- Alamat (URL): http://202.153.132.126/siduk/rpc.php
- Keterangan: Nilai kembalian jika valid
REST:<?xml version="1.0"><valid response><id>/id><name>/name><sex>/sex><horn>/horn><birth>/birth><address>/address><district>/district><town>/town
- Format Masukan: Text plain format
- Data Masukan: nik=1234567890123456
- Format Keluaran/Hasil: XML format
- Publikasi:

Buttons "Simpan" and "Batal" are located at the bottom right of the form. A footer at the bottom of the page reads "MODEL INTEROPERABILITAS SISTEM INFORMASI STUDI KASUS E-GOVERNMENT".

Gambar 5.8. Layar pengisian resource baru untuk pencarian seorang penduduk

Nama resource adalah nama sumberdaya informasi yang akan dipetakan dalam Web Services atau berkas RPC dalam hal ini bernama *seorang-penduduk*, Alamat URL adalah alamat lokasi RPC berada, Format Masukan dapat berupa teks biasa atau format XML, Format Keluaran dapat berupa teks biasa, format HTML, atau format XML. Data masukan dapat diisi dengan mengacu pada

Format Masukan. Data sumberdaya informasi yang diisi dapat langsung dipublikasikan dengan memilih kotak centang pada pilihan status Publikasi. Proses penyimpanan form tersebut dilakukan dengan menekan tombol Simpan. Berhasilnya proses penyimpanan tersebut mengakibatkan daftar katalog sumberdaya menjadi bertambah seperti terlihat pada Gambar 5.9.

No.	Penyedia	Nama Resource	Alamat (URL)	Format Masukan	Format Hasil	Keterangan	Publikasi	Pendata	Terdata	Ralat	Hapus	Pilih
1	depdagri	penduduk	http://202.153.132.116/siduk/rpc.php	txt	xml	Fungsi untuk mendapatkan data kependudukan berdasarkan NIK perorangan	<input checked="" type="checkbox"/>	indra	2009-07-07 00:14:55			<input type="checkbox"/>

Gambar 5.9. Layar daftar resource hasil penambahan baru

Gambar 5.10. Layar pengisian resource baru untuk pencarian wajib pajak

Perubahan status publikasi selain melalui proses ralat dapat juga dilakukan dengan mencentang kolom pilih dalam daftar untuk pilihan lebih dari satu sumberdaya informasi (resource), dilanjutkan dengan menekan tombol Aktif atau Pasif. Bentuk gambar kotak centang yang terdapat pada kolom Publikasi akan berpengaruh terhadap penekanan tombol Aktif dan Pasif apabila ada centangan pada kolom Pilih. Akibat dari perubahan status Publikasi tersebut berpengaruh pada daftar sumberdaya informasi yang ditampilkan aplikasi Resources Discovery, jika statusnya tidak terpublikasi maka sumberdaya informasi tersebut tidak akan ditampilkan dalam daftar Resource Discovery, begitupun sebaliknya.


Hasil pengujian dari proses registrasi sumberdaya informasi ini adalah berhasilnya proses registrasi yang dilakukan melalui aplikasi Resources Registry, sehingga data tersebut dapat dikelola untuk publikasi sumberdaya informasi sesuai dengan konsep Model Interoperabilitas Sistem Informasi melalui arsitektur yang berorientasi pada sumberdaya Informasi (ROA).

2. Pengujian penemuan/pencarian sumberdaya informasi

Resources Discovery merupakan aplikasi yang perlu diuji kemampuannya dalam memberikan fasilitas kepada publik atau Requester untuk menemukan sumberdaya informasi. Aplikasi memiliki tiga jenis konten yaitu konten daftar sumberdaya informasi, konten rincian sumberdaya informasi, dan konten percobaan transaksi sumberdaya informasi. Konten daftar sumberdaya informasi memiliki fasilitas penemuan atau pencarian berdasarkan nama inisial penyedia (Provider) dan nama sumberdaya informasi (resource). Pencarian dapat dilakukan pada salah satu pilihan atau pada sebagian kata, misalnya pencarian berdasarkan penyedia yang memiliki kata depan depda. Bentuk konten ini ditampilkan dalam Gambar 5.11.

Untuk melihat rincian deskripsi suatu sumberdaya informasi yang disediakan oleh provider misalnya *seorang-penduduk*, gambar yang terdapat pada kolom keterangan bisa diklik sehingga akan tampil layar yang berisi rincian deskripsi sumberdaya informasi sebagai referensi para Requester dalam menyusun proses pengolahan data dalam fungsi interface setelah mendapatkan data dari Web

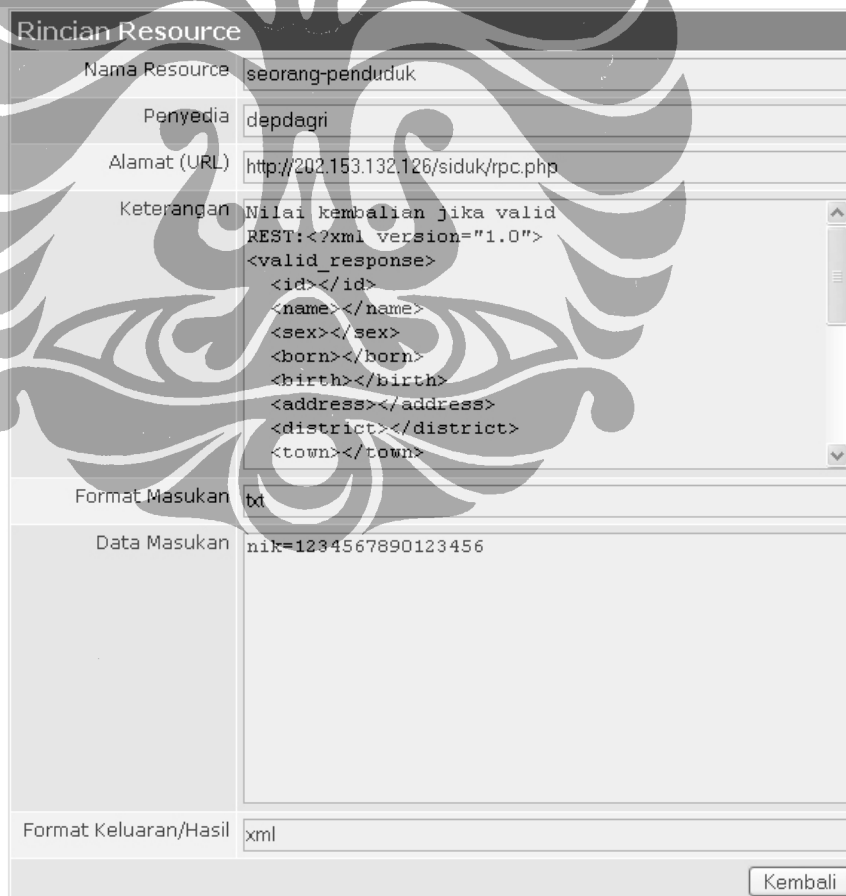
Services. Bentuk layar konten rincian sumberdaya informasi untuk mengambil data *seorang-penduduk* terdapat pada Gambar 5.12.



The screenshot shows a web browser window titled "Resources Discovery" with the URL "http://202.153.132.116/resdiscovery/". Below the browser window is a table listing resources.

No.	Penyedia	Nama Resource	Format Masukan	Format Hasil	Tercatat	Keterangan	Coba
1	depdagri	beberapa-penduduk	txt	xml	2009-04-15 00:26:58		
2	depdagri	lembaran-penduduk	txt	xml	2009-04-15 00:33:18		
3	depdagri	seorang-penduduk	txt	xml	2009-04-15 00:14:09		
4	depkeu	beberapa-wajibpajak	txt	xml	2009-04-15 00:40:55		
5	depkeu	lembaran-wajibpajak	txt	xml	2009-04-15 00:44:38		

Gambar 5.11. Layar daftar resource yang terpublikasi



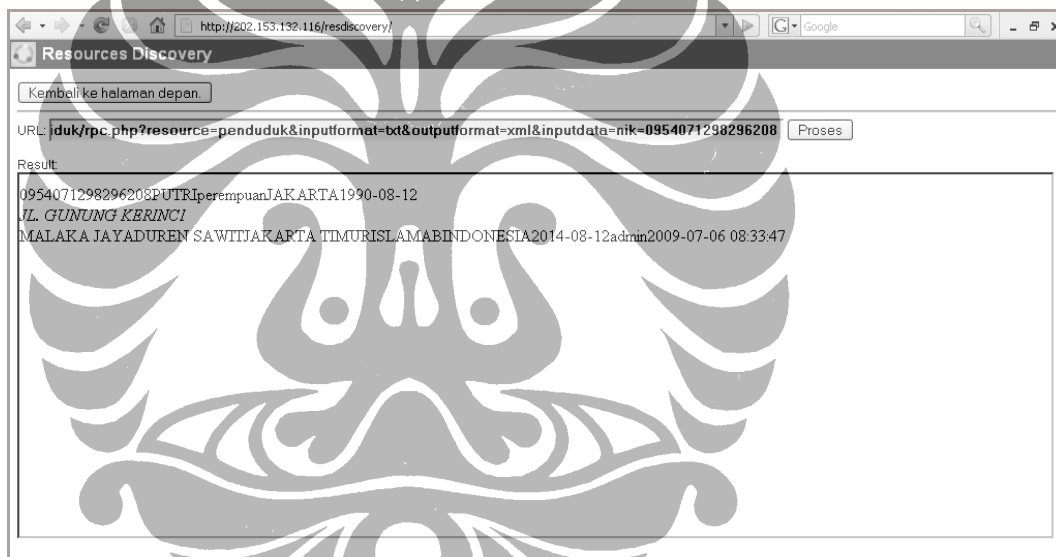
The screenshot shows the "Rincian Resource" page for the resource "seorang-penduduk". The page displays the following details:

- Nama Resource:** seorang-penduduk
- Penyedia:** depdagri
- Alamat (URL):** http://202.153.132.126/siduk/rpc.php
- Keterangan:** Nilai kembalian jika valid
REST:<?xml version="1.0">
<valid_response>
<id></id>
<name></name>
<sex></sex>
<born></born>
<birth></birth>
<address></address>
<district></district>
<town></town>
- Format Masukan:** txt
- Data Masukan:** nik=1234567890123456
- Format Keluaran/Hasil:** xml

A "Kembali" (Back) button is located at the bottom right of the page.

Gambar 5.12. Layar rincian deskripsi resource

Konten percobaan transaksi sumberdaya informasi pada Web Services dapat dilakukan dengan menekan tombol bergambar roda mesin yang terdapat dalam kolom Coba. Fasilitas yang ada pada layar ini adalah untuk menguji baris perintah URL yang dilengkapi dengan parameter input data dalam bentuk (URI) untuk diproses dalam kotak internal frame (iframe), apabila tombol proses ditekan maka akan terlihat hasilnya pada kotak internal frame. Berhasil tidaknya transaksi sangat tergantung pada format dan nilai masukan yang dilewatkan ke dalam parameter URL. Contoh kasus pada sumberdaya informasi kependudukan dapat dilakukan dengan memilih percobaan transaksi pada baris data dengan nama resource *seorang-penduduk* untuk mencari data seorang penduduk melalui NIK. Bentuk hasil dari transaksi konten ini ditampilkan dalam Gambar 5.13.



Gambar 5.13. Layar transaksi percobaan Web Services

3. Pengujian pertukaran informasi antar Sistem Informasi

Pengujian akhir ini merupakan pengujian yang dapat menentukan apakah model interoperabilitas sistem informasi layanan publik ini dapat diterapkan dalam pertukaran atau pengumpulan sumberdaya informasi antar sistem informasi yang berbeda.

Pada contoh kasus e-Government ini, diasumsikan pihak Sistem Informasi Perpajakan hanya akan mengambil informasi seorang wajib pajak dari data yang tersedia dalam Sistem Informasi Kependudukan berdasarkan NIK yang terdapat

pada KTP seseorang saat membuat NPWP baru. Melalui publikasi sumberdaya informasi yang dilakukan pihak sistem informasi kependudukan dalam memberikan layanan publik untuk memperoleh informasi penduduk melalui Web Services, maka pihak perpajakan dapat mengunduh perintah URL dari website Resources Discovery yang disediakan oleh Agent/Broker Publik.

Setelah mengunduh dan memasukan perintah URL dalam fungsi interface Sistem Informasi Perpajakan sebagaimana telah dijelaskan pada tahap implementasi sebelumnya maka proses pembuatan NPWP baru, melalui aplikasi perpajakan dapat diuji dengan diawali pada proses penambahan Wajib Pajak Perorangan yang terdapat dalam menu daftar wajib pajak. Sebelum proses validasi dan pengambilan data penduduk dilakukan, data penduduk dilengkapi lebih dahulu melalui proses penambahan data penduduk yang akan menghasilkan NIK secara otomatis setelah penyimpanan dilakukan. Langkah tersebut dilakukan dengan menekan tombol Tambah pada layar daftar penduduk Sistem Informasi Kependudukan, seperti terlihat pada tampilan Gambar 5.14, dan Gambar 5.15.



No.	NIK	Nama	Tgl.Lahir	Kelurahan	Kecamatan	Kota	Tgl.Pengurusan	Batas Berlaku	Ralat	Hapus	Pilih
1	0954071105888404	AGUS	05-Apr-1980	PULOMAS	KELAPA GADING	JAKARTA TIMUR	06-Jul-2009	05-Apr-2014			<input type="checkbox"/>
2	0954071298296208	PUTRI	12-Aug-1990	MALAKA JAYA	DUREN SAWIT	JAKARTA TIMUR	06-Jul-2009	12-Aug-2014			<input type="checkbox"/>
3	0954071327150944	RIDA	19-Jan-1988	CIPINANG	JATINEGARA	JAKARTA TIMUR	07-Jul-2009	19-Jan-2014			<input type="checkbox"/>
4	0954071371161567	DIDI SUKYADI	16-Feb-1973	KLENDER	DUREN SAWIT	JAKARTA TIMUR	09-Jul-2009	16-Feb-2014			<input type="checkbox"/>
5	0954072565585239	ERIK	19-Mar-1982	TEBET	PANCOBAN	JAKARTA SELATAN	15-Apr-2009	19-Mar-2014			<input type="checkbox"/>
6	0954073079235013	FERDINAN	18-Jun-1977	CEMPAKA PUTIH	RAWASARI	JAKARTA PUSAT	15-Apr-2009	18-Jun-2014			<input type="checkbox"/>
7	0954073495104664	WINDA	12-Sep-1976	MALAKA JAYA	DUREN SAWIT	JAKARTA TIMUR	15-Apr-2009	12-Sep-2014			<input type="checkbox"/>
8	0954074809546953	ARIS	18-Aug-1983	KEMANG	CIPETE	JAKARTA SELATAN	15-Apr-2009	18-Aug-2014			<input type="checkbox"/>
9	0954074876109403	LINDA	19-Feb-1978	BATU TULIS	CIPAYUNG	JAKARTA SELATAN	15-Apr-2009	19-Feb-2014			<input type="checkbox"/>
10	0954074942306887	DESI	10-Jan-1979	RAWAMANGUN	PULO GADUNG	JAKARTA TIMUR	15-Apr-2009	10-Jan-2014			<input type="checkbox"/>

Gambar 5.14. Layar daftar penduduk pada Sistem Informasi Kependudukan

Penduduk Baru	
Nama	IRWANSYAH
Jenis Kelamin	Laki-laki
Tempat Lahir	BANDUNG
Tanggal Lahir	20/3/1978 (hari/bulan/tahun)
Alamat	JL. GARUDA
Kelurahan	ANGKASA
Kecamatan	KEMAYORAN
Kota	JAKARTA PUSAT
Agama	ISLAM
Golongan Darah	A
Warga Negara	INDONESIA
<input type="button" value="Simpan"/> <input type="button" value="Batal"/>	

Gambar 5.15. Layar registrasi penduduk baru.

Pada Gambar 5.14, diilustrasikan daftar penduduk yang tersimpan dalam database sistem informasi kependudukan. Dari sekian banyak penduduk dipilih NIK salah satu orang yang dianggap akan mendaftarkan diri menjadi wajib pajak pada dinas perpajakan. Pilihannya adalah penduduk dengan nama PUTRI yang memiliki NIK dengan nomor 0954071298296208.

Kemudian Sistem Informasi Perpajakan dijalankan dengan mengakses alamat <http://202.153.132.116/sijak>. Melalui pilihan menu Wajib Pajak daftar wajib pajak perorangan ditampilkan sepuluh baris perhalaman. Pencarian wajib pajak berdasarkan NIK dilakukan sebagai verifikasi awal untuk menjamin penduduk tersebut belum terdaftar sebagai wajib pajak. Apabila berdasarkan hasil pencarian ternyata belum terdaftar, maka proses penambahan wajib pajak baru dapat dilakukan dengan menekan tombol Tambah dan akan menampilkan layar pengisian data wajib pajak seperti dalam Gambar 5.16.

Wajib Pajak Baru	
NIK	0954071298296208 <input type="button" value="Validasi"/>
Nama	<input type="text"/>
Tempat Lahir	<input type="text"/>
Tanggal Lahir	<input type="text"/> (hari/bulan/tahun)
Alamat	<input type="text"/>
Kelurahan	<input type="text"/>
Kecamatan	<input type="text"/>
Kota	<input type="text"/>
Lapangan Usaha	Pilih <input type="button" value="v"/>
Status Modal	Pilih <input type="button" value="v"/>
PPh Pasal 4	<input type="checkbox"/>
PPh Pasal 15	<input checked="" type="checkbox"/>
PPh Pasal 19	<input type="checkbox"/>
PPh Pasal 21	<input type="checkbox"/>
PPh Pasal 22	<input type="checkbox"/>
PPh Pasal 23	<input type="checkbox"/>
PPh Pasal 25	<input type="checkbox"/>
PPh Pasal 26	<input type="checkbox"/>
PPh Pasal 29	<input type="checkbox"/>
<input type="button" value="Simpan"/> <input type="button" value="Batal"/>	

Gambar 5.16. Layar registrasi wajib pajak baru sebelum validasi

Data pertama yang perlu diisi adalah NIK berdasarkan KTP, setelah mengisi NIK proses validasi dan pengambilan data dari sistem informasi kependudukan dilakukan dengan menekan tombol Validasi. Jika proses validasi berhasil maka sebagian data penduduk tersebut akan mengisi kolom data yang

tersedia dibawahnya, seperti Nama, Tempat Lahir, Tanggal Lahir, Alamat sampai dengan Kota. Hasil proses ini dapat dilihat pada Gambar 5.17.

Wajib Pajak Baru	
NIK	0954071298296208 <input type="button" value="Validasi"/> Lama proses: 0,006 detik
Nama	PUTRI
Tempat Lahir	JAKARTA
Tanggal Lahir	12/08/1990 (hari/bulan/tahun)
Alamat	JL. GUNUNG KERINCI
Kelurahan	MALAKA JAYA
Kecamatan	DUREN SAWIT
Kota	JAKARTA TIMUR
Lapangan Usaha	Karyawan Swasta <input type="button" value="v"/>
Status Modal	Swasta <input type="button" value="v"/>
PPh Pasal 4	<input type="checkbox"/>
PPh Pasal 15	<input type="checkbox"/>
PPh Pasal 19	<input type="checkbox"/>
PPh Pasal 21	<input checked="" type="checkbox"/>
PPh Pasal 22	<input type="checkbox"/>
PPh Pasal 23	<input type="checkbox"/>
PPh Pasal 25	<input type="checkbox"/>
PPh Pasal 26	<input type="checkbox"/>
PPh Pasal 29	<input type="checkbox"/>
<input type="button" value="Simpan"/> <input type="button" value="Batal"/>	

Gambar 5.17. Layar registrasi PUTRI sebagai wajib pajak baru setelah validasi.

Dengan demikian proses validasi dan pengambilan data penduduk berhasil dilakukan, sebaliknya jika tidak berhasil maka kolom tersebut akan tetap kosong. Selain itu validasi data penduduk yang dilakukan melalui Web Services dari Sistem Informasi Perpajakan telah memakan waktu proses kurang lebih selama

0,005 sampai 0,006 detik per 1 baris data penduduk. Proses seperti ini dapat dilakukan berkali-kali pada penduduk yang belum terdaftar menjadi wajib pajak, dengan syarat data penduduk tersebut sudah terdaftar lebih dulu dalam data Sistem Informasi Kependudukan.

Untuk mendapatkan hasil yang meyakinkan pada pengujian waktu proses maka dilakukan proses registrasi wajib pajak sebanyak tiga kali, hingga mendapatkan informasi lebih banyak yang sama pada penghitungan waktu pengambilan data melalui Web Services dalam Sistem Informasi Kependudukan. Hasil perhitungan waktu proses tersebut diperlihatkan pada Gambar 5.18. dan Gambar 5.19.



The image shows a web form titled "Wajib Pajak Baru" (New Taxpayer Registration). The form contains the following fields and values:

Field	Value
NIK	0954073079235013
Nama	FERDINAN
Tempat Lahir	KUPANG
Tanggal Lahir	18/06/1977 (hari/bulan/tahun)
Alamat	JL. CEMPAKA PUTIH BARAT
Kelurahan	CEMPAKA PUTIH
Kecamatan	RAWASARI
Kota	JAKARTA PUSAT
Lapangan Usaha	Pilih
Status Modal	Pilih
PPh Pasal 4	<input type="checkbox"/>
PPh Pasal 15	<input type="checkbox"/>
PPh Pasal 19	<input type="checkbox"/>
PPh Pasal 21	<input type="checkbox"/>
PPh Pasal 22	<input type="checkbox"/>
PPh Pasal 23	<input type="checkbox"/>
PPh Pasal 25	<input type="checkbox"/>
PPh Pasal 26	<input type="checkbox"/>
PPh Pasal 29	<input type="checkbox"/>

Buttons: Validasi, Simpan, Batal. Performance: Lama proses: 0,005 detik.

Gambar 5.18. Registrasi FERDINAN sebagai wajib pajak baru setelah validasi.

Pada proses kali ini menunjukkan hasil perhitungan waktu proses yang berbeda dengan proses sebelumnya dilakukan untuk registrasi wajib pajak atas nama PUTRI. Untuk meyakinkan peneliti, maka proses tersebut diulang kembali terhadap penduduk yang bernama LINDA.

Wajib Pajak Baru	
NIK	0954074876109403 <input type="button" value="Validasi"/> Lama proses: 0,006 detik
Nama	LINDA
Tempat Lahir	JAKARTA
Tanggal Lahir	19/02/1978 (hari/bulan/tahun)
Alamat	JL MERAPI
Kelurahan	BATU TULIS
Kecamatan	CIPAYUNG
Kota	JAKARTA SELATAN
Lapangan Usaha	Pilih <input type="button" value="v"/>
Status Modal	Pilih <input type="button" value="v"/>
PPh Pasal 4	<input type="checkbox"/>
PPh Pasal 15	<input type="checkbox"/>
PPh Pasal 19	<input type="checkbox"/>
PPh Pasal 21	<input type="checkbox"/>
PPh Pasal 22	<input type="checkbox"/>
PPh Pasal 23	<input type="checkbox"/>
PPh Pasal 25	<input type="checkbox"/>
PPh Pasal 26	<input type="checkbox"/>
PPh Pasal 29	<input type="checkbox"/>
<input type="button" value="Simpan"/> <input type="button" value="Batal"/>	

Gambar 5.19 Registrasi LINDA sebagai wajib pajak baru setelah validasi

Namun waktu prosesnya tidak begitu jauh dari hasil sebelumnya. Ini disebabkan proses pengambilan data hanya pada satu baris data saja. Hasil penambahan data wajib pajak tersebut dapat terlihat pada Gambar 5.20.

No.	NPWP	NIK	Nama	Tgl.Lahir	Kelurahan	Kecamatan	Kota	Tgl.Terdaftar	Ralat	Hapus	Pilih
1	192661066378470	0954079783302820	IRWANSYAH	20-Mar-1978	ANGKASA	KEMAYORAN	JAKARTA PUSAT	15-Apr-2009			<input type="checkbox"/>
2	192663784706811	0954071298296208	PUTRI	12-Aug-1990	MALAKA JAYA	DUREN SAWIT	JAKARTA TIMUR	15-Apr-2009			<input type="checkbox"/>
3	192663915561379	0954071105838404	AGUS	05-Apr-1980	PULOMAS	KELAPA GADING	JAKARTA TIMUR	15-Apr-2009			<input type="checkbox"/>
4	192664961431972	0954074809546953	ARIS	18-Aug-1983	KEMANG	CIPETE	JAKARTA SELATAN	15-Apr-2009			<input type="checkbox"/>
5	192666934436676	0954074942306887	DESI	10-Jan-1979	RAWAMANGUN	PULO GADUNG	JAKARTA TIMUR	15-Apr-2009			<input type="checkbox"/>
6	192668310513648	0954071371161567	DIDI SUKYADI	16-Feb-1973	KLENDER	DUREN SAWIT	JAKARTA TIMUR	15-Apr-2009			<input type="checkbox"/>
7	192668806661572	0954072565585239	ERIK	19-Mar-1982	TEBET	PANCORAN	JAKARTA SELATAN	15-Apr-2009			<input type="checkbox"/>

Gambar 5.20. Layar daftar wajib pajak baru.

Cara lain yang dapat dilakukan untuk melakukan registrasi wajib pajak adalah dengan mengambil beberapa data penduduk berdasarkan kriteria sebagian NIK atau Nama penduduk melalui menu Penduduk pada Sistem Informasi Perpajakan, kemudian dari daftar penduduk yang tayang pada layar aplikasi seluruh atau sebagian penduduk dalam daftar tersebut dapat dipilih untuk registrasi wajib pajak. Setelah proses registrasi dilakukan melalui penekanan tombol register, maka pada layar akan tampil NPWP dari masing-masing data penduduk yang terpilih sebelumnya. Ilustrasi dari proses ini terdapat pada Gambar 5.21, Gambar 5.22, dan Gambar 5.23.

Hasil proses pada Gambar 5.21, dan Gambar 5.22, menunjukkan lamanya proses pengambilan data penduduk melalui Web Services Sistem Informasi Kependudukan, yaitu sekitar 0,007 detik per 10 baris data penduduk dan sekitar 0,005 detik hingga 0,006 detik per 2 baris data penduduk. Sedangkan lamanya proses registrasi pada Gambar 5.23, memakan waktu sekitar 0,007 detik untuk proses registrasi terhadap 5 data penduduk ke dalam database lokal, namun proses pengambilan data penduduk melalui Web Services Sistem Informasi Kependudukan memakan waktu 0,006 detik untuk 10 baris data penduduk. Hasil ini akan terlihat perbedaannya jika dilakukan pada transaksi data lebih dari 50 baris data. Namun karena aplikasi ini membagi pengolahan data dalam beberapa segmen halaman data, maka hal ini belum dapat dilihat perbedaannya.

Alasan utama dilakukan proses per halaman karena proses antara browser dengan web server akan memakan waktu lebih lama ketimbang transaksi data

antara Sistem Aplikasi dengan Web Services di lokasi server yang memiliki performance cukup tinggi dan bandwidth akses internet yang cukup besar sekitar 1Mbps hingga 4Mbps, sementara akses internet browser client umumnya hanya memiliki kecepatan hingga 64Kbps pada koneksi modem.

No.	NIK	NPWP	Nama	Tgl.Lahir	Kelurahan	Kecamatan	Kota	Tgl.Pengurusan	Batas Berlaku	Pilih
1	0954071105838404		AGUS	05-Apr-1980	PULOMAS	KELAPA GADING	JAKARTA TIMUR	06-Jul-2009	05-Apr-2014	<input type="checkbox"/>
2	0954071298296208		PUTRI	12-Aug-1990	MALAKA JAYA	DUREN SAWIT	JAKARTA TIMUR	06-Jul-2009	12-Aug-2014	<input type="checkbox"/>
3	0954071327150944		RIDA	19-Jan-1988	CIPINANG	JATINEGARA	JAKARTA TIMUR	07-Jul-2009	19-Jan-2014	<input type="checkbox"/>
4	0954071371161567		DIDI SUKYADI	16-Feb-1973	KLENDER	DUREN SAWIT	JAKARTA TIMUR	02-Jul-2009	16-Feb-2014	<input type="checkbox"/>
5	0954072565585239		ERIK	19-Mar-1982	TEBET	PANCORAN	JAKARTA SELATAN	15-Apr-2009	19-Mar-2014	<input type="checkbox"/>
6	0954073079235013		FERDINAN	18-Jun-1977	CEMPAKA PUTIH	RAWASARI	JAKARTA PUSAT	15-Apr-2009	18-Jun-2014	<input type="checkbox"/>
7	0954073495104664		WINDA	12-Sep-1976	MALAKA JAYA	DUREN SAWIT	JAKARTA TIMUR	15-Apr-2009	12-Sep-2014	<input type="checkbox"/>
8	0954074809546953		ARIS	18-Aug-1983	KEMANG	CIPETE	JAKARTA SELATAN	15-Apr-2009	18-Aug-2014	<input type="checkbox"/>
9	0954074876109403		LINDA	19-Feb-1978	BATU TULIS	CIPAYUNG	JAKARTA SELATAN	15-Apr-2009	19-Feb-2014	<input type="checkbox"/>
10	0954074942306887		DESI	10-Jan-1979	RAYAMANGUN	PULO GADUNG	JAKARTA TIMUR	15-Apr-2009	10-Jan-2014	<input type="checkbox"/>

Gambar 5.21. Layar daftar penduduk halaman 1.

No.	NIK	NPWP	Nama	Tgl.Lahir	Kelurahan	Kecamatan	Kota	Tgl.Pengurusan	Batas Berlaku	Pilih
11	0954079445430638		LUSTIANA	12-Oct-1980	MALAKA SARI	DUREN SAWIT	JAKARTA TIMUR	15-Apr-2009	12-Oct-2014	<input type="checkbox"/>
12	0954079783302820	192661066378470	IRWANSYAH	20-Mar-1978	ANGKASA	KEMAYORAN	JAKARTA PUSAT	15-Apr-2009	20-Mar-2014	<input type="checkbox"/>

Gambar 5.22. Layar daftar penduduk halaman 2.

http://202.153.132.116/sisak/

Sistem Informasi Perpajakan Pengguna : ADMINISTRAT

Serambi Wajib Pajak Penduduk

NIK: Nama:

Halaman: << < 1 2 > >> Proses:

Total: 2 Halaman (12 Baris)

Registrasi berhasil dilakukan sebanyak 5 records selama **0,007** detik.

Data penduduk dimuat melalui Web Services Kependudukan selama **0,006** detik.

No.	NIK	NPWP	Nama	Tgl.Lahir	Kelurahan	Kecamatan	Kota	Tgl.Pengurusan	Batas Berlaku	Pilih
1	0954071105838404	192663915561379	AGUS	05-Apr-1980	PULOMAS	KELAPA GADING	JAKARTA TIMUR	06-Jul-2009	05-Apr-2014	<input type="checkbox"/>
2	0954071298296208	192663784706811	PUTRI	12-Aug-1990	MALAKA JAYA	DUREN SAWIT	JAKARTA TIMUR	06-Jul-2009	12-Aug-2014	<input type="checkbox"/>
3	0954071327150944		RIDA	19-Jan-1988	CIPINANG	JATINEGARA	JAKARTA TIMUR	07-Jul-2009	19-Jan-2014	<input type="checkbox"/>
4	0954071371161567	192668310513648	DIDI SUKYADI	16-Feb-1973	KLENDER	DUREN SAWIT	JAKARTA TIMUR	02-Jul-2009	16-Feb-2014	<input type="checkbox"/>
5	0954072565585239	192668806661572	ERIK	19-Mar-1982	TEBET	PANCORAN	JAKARTA SELATAN	15-Apr-2009	19-Mar-2014	<input type="checkbox"/>
6	0954073079235013		FERDINAN	18-Jun-1977	CEMPAKA PUTIH	RAWASARI	JAKARTA PUSAT	15-Apr-2009	18-Jun-2014	<input type="checkbox"/>
7	0954073495104664		WINDA	12-Sep-1976	MALAKA JAYA	DUREN SAWIT	JAKARTA TIMUR	15-Apr-2009	12-Sep-2014	<input type="checkbox"/>
8	0954074809546953	192664961431972	ARIS	18-Aug-1983	KEMANG	CIPETE	JAKARTA SELATAN	15-Apr-2009	18-Aug-2014	<input type="checkbox"/>
9	0954074876109403		LINDA	19-Feb-1978	BATU TULIS	CIPAYUNG	JAKARTA SELATAN	15-Apr-2009	19-Feb-2014	<input type="checkbox"/>
10	0954074942306887	192666934436676	DESI	10-Jan-1979	RAWAMANGUN	PULO GADUNG	JAKARTA TIMUR	15-Apr-2009	10-Jan-2014	<input type="checkbox"/>

Gambar 5.23. Layar proses hasil registrasi wajib pajak dari data penduduk.

Untuk membuktikan proses pertukaran informasi pada Model Interoperabilitas Sistem Informasi Layanan Publik dapat dilakukan adalah dengan memberikan akses kepada publik untuk mengetahui beberapa penduduk yang sudah terdaftar sebagai wajib pajak. Proses ini dilakukan melalui Sistem Informasi Kependudukan yang berstatus sebagai Requester terhadap Web Services Sistem Informasi Perpajakan sebagai Provider. Hasil proses pengambilan informasi data wajib pajak melalui Web Services Sistem Informasi Perpajakan terlihat pada Gambar 5.24. Waktu yang ditempuh adalah 0,006 detik per 7 baris data wajib pajak.

http://202.153.132.126/siduk/

Sistem Informasi Kependudukan Pengguna : AD

Serambi Penduduk Wajib Pajak

NIK: NPWP: Nama:

Halaman: << < > >>

Total: Halaman (Baris)

Data wajib-pajak dimuat melalui Web Services Perpajakan selama **0,006** detik.

No.	NIK	NPWP	Nama	Tgl.Lahir	Kelurahan	Kecamatan	Kota	Tgl.Terdarar
1	0954079783302820	192661066378470	IRWANSYAH	20-Mar-1978	ANGKASA	KEMAYORAN	JAKARTA PUSAT	15-Apr-2009
2	0954071298296208	192663784706811	PUTRI	12-Aug-1990	MALAKA JAYA	DUREN SAWIT	JAKARTA TIMUR	15-Apr-2009
3	0954071105838404	192663915561379	AGUS	05-Apr-1980	PULOMAS	KELAPA GADING	JAKARTA TIMUR	15-Apr-2009
4	0954074809546953	192664961431972	ARIS	18-Aug-1983	KEMANG	CIPETE	JAKARTA SELATAN	15-Apr-2009
5	0954074942306887	192666934436676	DESI	10-Jan-1979	RAWAMANGUN	PULO GADUNG	JAKARTA TIMUR	15-Apr-2009
6	0954071371161567	192668310513648	DIDI SUKYADI	16-Feb-1973	KLENDER	DUREN SAWIT	JAKARTA TIMUR	15-Apr-2009
7	0954072565585239	192668806661572	ERIK	19-Mar-1982	TEBET	PANCORAN	JAKARTA SELATAN	15-Apr-2009

Gambar 5.24. Layar daftar wajib pajak dari Sistem Informasi Perpajakan.

5.3. ANALISA HASIL PENGUJIAN

Berdasarkan proses pengujian terhadap Model Interoperabilitas Sistem Informasi Layanan Publik pada penelitian ini, menunjukkan tingkat validitas pada beberapa bagian pengujian sebagai berikut:

1. Pengujian registrasi publikasi sumberdaya informasi, dilakukan terhadap aplikasi Resources Registry seperti yang terlihat pada Gambar 5.5., Gambar 5.6, Gambar 5.7., Gambar 5.8, Gambar 5.9., dan Gambar 5.10. menunjukkan kemampuan aplikasi yang bersifat user friendly, integrateable dan scalable.
2. Pengujian penemuan sumberdaya informasi, ditujukan terhadap aplikasi Resource Discovery seperti yang terlihat pada Gambar 5.11., Gambar 5.12., dan Gambar 5.13. menunjukkan kemampuan aplikasi yang bersifat interoperable, scalable, user friendly, dan integrateable.
3. Pengujian pertukaran informasi antar Sistem Informasi, diuji pada dua Web Services yaitu untuk informasi penduduk dan wajib pajak. Seperti yang terlihat pada Gambar 5.16., Gambar 5.17., Gambar 5.18., Gambar 5.19., Gambar 5.10., Gambar 5.21., Gambar 5.22., dan Gambar 5.23. menunjukkan kemampuan aplikasi Web Services Penduduk yang bersifat interoperable, scalable, dan integrateable. Begitu pula pada aplikasi Web Services Wajib Pajak seperti yang terlihat pada Gambar 5.24. menunjukkan kemampuan aplikasi yang bersifat interoperable, scalable, dan integrateable.

Berdasarkan standar kebutuhan yang harus dipenuhi pada sebuah sistem aplikasi e-Government, hasil pengujian validitas tersebut tergambar dalam Tabel 5.4.

Analisa dari hasil pengujian tersebut menghasilkan beberapa kesimpulan yang meliputi:

1. Model Interoperabilitas Sistem Informasi Layanan Publik sudah dapat diimplementasikan sesuai prinsip-prinsip interoperabilitas melalui teknologi Web Services.
2. Kejadian selama proses uji coba aplikasi menunjukkan kemungkinan implementasi ini dapat digunakan untuk pengembangan yang bebas dari ketergantungan pada vendor.

3. Terhadap studi kasus dalam e-Government, implementasi ini dapat pula dijadikan acuan bagi pengembangan sistem aplikasi layanan publik sesuai kebijakan e-Government.

Tabel 5.4. Hasil pengujian Sistem Aplikasi pada Model Interoperabilitas Sistem Informasi Layanan Publik

Sistem Aplikasi	Interoperable		Scalable		User Friendly		Integrateable	
	Valid	Invalid	Valid	Invalid	Valid	Invalid	Valid	Invalid
Resource Registry			V		V		V	
Resource Discovery	V		V		V		V	
Web Services Penduduk	V		V				V	
Web Services Wajib pajak	V		V				V	

